

MASSACHUSETTS INSTITUTE OF TECHNOLOGY  
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 601

November, 1980

CONCLUSIONS FROM THE COMMODITY EXPERT PROJECT

James L. Stansfield

ABSTRACT. The goal of the commodity expert project was to develop a prototype program that would act as an intelligent assistant to a commodity market analyst. Since expert analysts must deal with very large, yet incomplete, data bases of unreliable facts about a complex world, the project would stringently test the applicability of Artificial Intelligence techniques. After a significant effort however, I am forced to the conclusion that an intelligent, real-world system of the kind envisioned is currently out of reach. Some of the difficulties were due to the size and complexity of the domain. As its true scale became evident, the available resources progressively appeared less adequate. The representation and reasoning problems that arose were persistently difficult and fundamental work is needed before the tools will be sufficient to engineer truly intelligent assistants. Despite these difficulties, perhaps even because of them, much can be learned from the project. To assist future applications projects, I explain in this report some of the reasons for the negative result, and also describe some positive ideas that were gained along the way. In doing so, I hope to convey the respect I have developed for the complexity of real-world domains, and the difficulty of describing the ways experts deal them.

Acknowledgements: This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-80-C-0505.

© MASSACHUSETTS INSTITUTE OF TECHNOLOGY 1980

## Introduction

When the commodities assistant project began three years ago, there was a feeling that Artificial Intelligence had emerged from the micro-world stage and could be applied to real domains in a more general way. The existence of several large application systems supported this belief. New techniques would possibly give rise to a generation of smarter, more general systems. LUNAR [Woods et al.] provided access to a large data base of facts about moon rocks and allowed a user to query this data base in a subset of English. The MYCIN program [Shortliffe] requested patient information from a doctor and diagnosed a small class of diseases. DENDRAL [Buchanan et al.] was a significant aid to experts in analyzing mass spectroscopy data. While these expert programs were under development, so were new "general purpose" tools for AI. Knowledge representation languages such as FRL [Goldstein & Roberts] and KRL [Bobrow & Winograd] were spawned by the work on Frames [Minsky], and there had been many developments in the area of Semantic Nets [Schubert] [Hendrix]. In order to implement reasoning, procedural deduction languages such as CONNIVER [Mcdermott & Sussman], and AMORD [de Kleer et al.] were developing from the earlier MICRO-PLANNER [Sussman et al.] work. For language understanding, a new parser was in preparation [Marcus] and much work was in progress on discourse [Schank] [Cullingford] [Rosenberg] and the problems of reference [Sidner] [Webber].

As if to challenge these developments, a new problem was recognized, and sometimes referred to as the fusion problem. In a typical fusion situation, an expert is supplied with a very rapid flow of information containing both numeric and verbal components. It is continually updated to reflect a changing state of affairs in the world, but some information may be missing and the expert may be misinformed. In this noisy situation, the expert has to analyze the data, watching for specific types of important situation and acting accordingly. As the flow of information increases, this becomes increasingly difficult so an important aspect of the fusion problem is to design a computer program that acts as an intelligent interface between the data and the expert. To be effective, such an intelligent assistant is likely to need considerable expertise itself. It must be able to interpret the information and communicate it to a naive user, and it must have some model of the user's requirements.

It was thought that the new tools of AI would help solve the fusion problem by providing versatile programs with a wide range of cognitive skills. The commodities project was designed to do just this in an economic domain. Commodities was viewed as ideal because it paralleled all the aspects of the fusion problem. An expert commodity analyst is faced with a mass of ever-changing data that emanates from the production and marketing system for commodities. Included are both facts and figures, and the information is often incomplete and in error. Real commodities experts continually try to analyze their data to produce summaries and forecasts. Commodities is clearly not a toy problem but is a real application and would put the new AI methods to the test. Initially, three people were involved in the project. Since two of these left after a year, this report gives my perspective on the project and the opinions are my own.

After expending considerable time and effort, and trying a variety of approaches, I conclude that the task as specified is far too ambitious. The commodities domain is huge and complex and my attempts to simplify it resulted in the toy worlds that I wished to avoid. While many of the problems were created by the sheer size of the domain and the complexity of the interactions between its parts, others arose because current representation languages were inadequate to capture particular subtleties that needed distinguishing within the program's knowledge. I also feel that necessary resources were missing. For example, a commodity expert must be available for consultation, and on-line data would have helped. Despite these difficulties, and perhaps even because of them, I feel that much can be learned from the negative result. In this report, I try to identify some of the reasons for the failure together with some of the positive ideas that were gained in the process. I hope this will be helpful in future applications projects and will suggest areas where further research is prerequisite. In working on the project, I have come to respect the complexity of the domains that human experts cope with and the difficulty of describing the ways they deal with them.

The next sections each describe and criticize a different approach I have taken to the commodities assistant problem. Since many of the questions that arose relate to representation issues, I had hoped to include a further section dealing directly with representation, and evaluating some of the methods and mechanisms provided by current "knowledge" representation languages.



Unfortunately, time constraints prevented this, though some discussion of representation issues is spread through the report as needed to support the argument. I currently feel that much work is still needed before we will be able to apply general representation and reasoning methods toward genuinely intelligent systems. Interim solutions to the fusion problem must use more specific techniques. Representations will need tailoring to specific tasks, and domain-dependant problem-solving procedures may be more appropriate than general reasoning methods. In particular, the purpose of an intelligent assistant project must be very clearly specified, and its scope carefully restricted.

### **Understanding commodities news articles.**

My first approach to the design of an intelligent assistant for a commodities analyst was based on a type of information that I call "commentary". Commentary is verbal news about events, and commodities commentary can be found in *The Wall Street Journal*, *The New York Times*, and *The Grain Market News*. I planned to design a computer program that would act as an assistant to an expert by digesting large amounts of commentary and presenting it in a more usable form. Such a program might have various abilities such as summarizing information, pruning out uninteresting details, keeping track of news about specified situations, explaining, alerting the expert to special types of situation, or even forecasting events. Since commodities commentary is available in machine-readable form from news wires, there was an attractive prospect of testing a prototype on "live" data. But in practice, the commodities domain proved far too complex for any general implementation of these skills. The language and representation problems alone are enormous and I will describe some of these difficulties in more detail.

An examination of several hundred news articles about grain showed that most of the commentary consisted of simple assertions about very common events. There was little discussion of their cause, their consequences, or their significance. Here are three examples.

"Rainfall in the Mid-west improved crop prospects."

"World corn export demand is slow."

"Corn harvest picked up momentum."

It should be possible to invent a representation language and vocabulary adequate to frame all of these simple sentences but it is unclear what would be gained in the process. The content of such sentences does not support very far-ranging inferences and they are very disconnected from each another. They raise far more questions than they answer. For example, how much rainfall was there in the Mid-west? Where did it fall? What was the state of the top-soil moisture and the condition of the crop? Even with answers to these questions, more information is needed to determine the kind and amount of crop improvement and how it would affect the market? The purpose of the articles in which these statements occurred is only to summarize the major items of note. There is little for an assistant to do.

In the articles I examined, besides reports about the most common events affecting supply and demand, it frequently happened that new types of event were introduced. As these were equally important in determining the market behavior and had to be included, the domain rapidly expanded without control. The lack of a natural cut-off point at which to close the domain became a significant problem. Three of these irregular events are shown next.

"Synfuel has been developed as a new use for corn."

"Farmers anticipate a change in the loan rate."

"The administration has been considering ordering a 10% cut in acreage planted for corn. A decision had been expected by now."

The third of these is a particularly good illustration of the complex influences that must be included in a realistic model of the commodities domain. It is hard to handle such influences in a program's analysis, especially since some effects are difficult even for people to allow for. Yet such crucial effects should not be excluded from the analysis if it is to reflect the real world well enough to be useful to an expert.

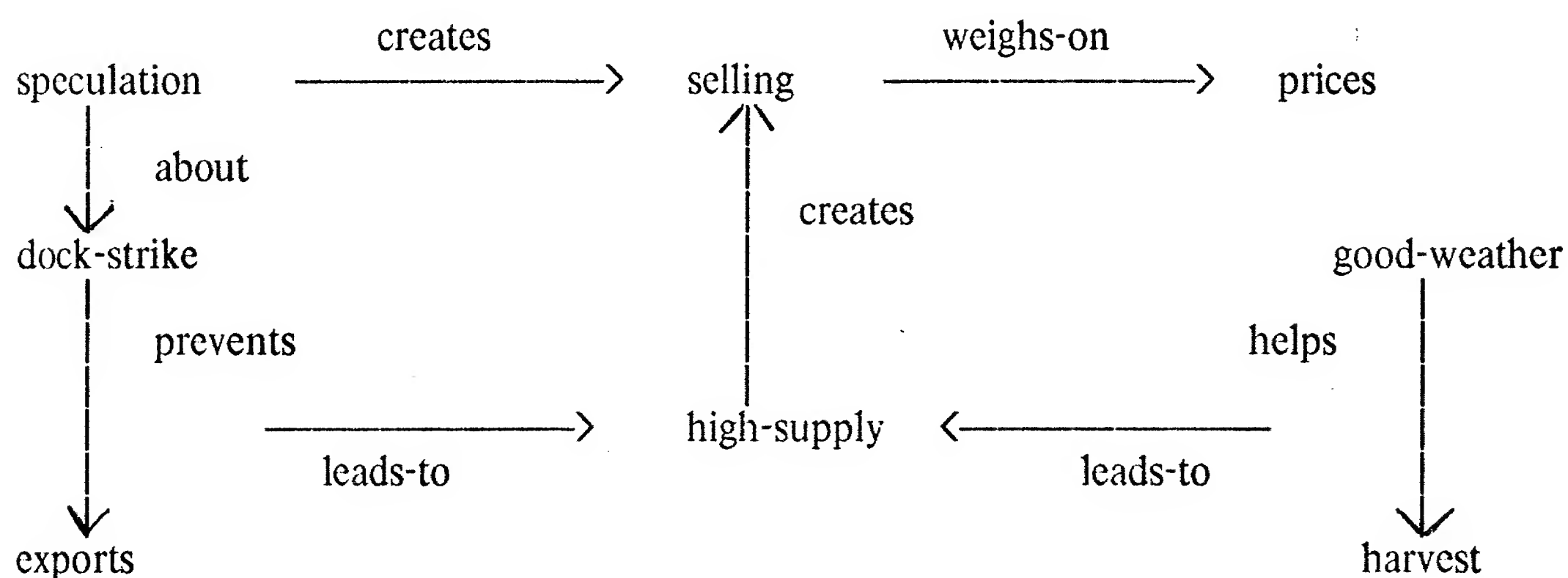
A third class of commentary includes sequences of statements that form short arguments. After encountering difficulties with the other kinds, it is tempting to use these as simple scenarios. A program might read one or two of them and use a limited amount of knowledge to make connecting inferences. This seems an attractive and gradual way to develop programs that understand natural language discourse. In fact, a more detailed look at an example shows that it is not so simple after

all.

"In futures markets, brokers said that speculation about spreading dockworker strike problems at grain exporting terminals continued to create selling that weighed on prices, as did forecasts of generally clear, harvest-enhancing weather across much of the Mid-west for the next several days."

Schematic representations of passages such as this can be constructed using a basic semantic net formalism and, in this work, I used my own modified version of FRL. Figure 1 shows a first pass at a representation and creates an illusion that the content of the passage, while not completely captured, has at least been well pinned down. The network's cursory quality makes it appear that significant inference steps have been represented and that only a sharpening of the representation is necessary. There is a misleading feeling that similar representations of ten or twenty short passages would make a significant dent into the problem of understanding commodities news articles, and that the process would be nicely cumulative. It is a serious mistake for, as I will show, this particular network is far from the solid and usable representation of the example passage that is needed. Investigating such fragments is surely one way to shed some light on representation issues but, in the near term, it leads away from the goal of designing an intelligent assistant. For that, we need a much broader and more detailed model of commodities and a more powerful representation in which to express it.

Fig. 1. A schematic representation of the passage.



The links in figure 1 are very imprecise statements of the causal relationships suggested by the passage. For example, "good weather helps the harvest" refers to a complicated event that must be described more fully before any consequences can be inferred. Its verb, "help", refers to a very general concept, and its specific instantiation in this passage requires considerable expansion. One possibility is that weather conditions allow the harvest to progress rapidly. The subject is not the good weather as it is actually farmers who are being helped. We must then determine the goal or process they are being helped with. The process is, of course, the harvest, and their goal is to complete it as rapidly as possible. Saying that this aid "leads-to" high supplies is also vague. Possibly, a rapid harvest would guard against any corn being damaged by bad weather. This would increase the expected production figures but need not lead to a high supply, simply to a higher supply than was previously expected. How does this create selling? Create suggests that the selling did not exist before, but it would be equally reasonable to have said "high-prices affected selling" which takes a different view. The link is really short-hand for something like the following. "The possibility of high supply led traders to think the price would become lower than the current futures market price. They sold more futures than they would otherwise have done and this weighed on futures prices though it was only one of the influences on futures trading."

Situations like this came up whenever I tried to represent news articles. It seemed that the expansion process described was a necessary step in utilizing the data. However, it was full of difficulties. A major problem was the lack of a good epistemology of words and concepts and an excellent example of this is in the representation of causality. An examination of many news articles showed that causality-related concepts occur as part of the meaning of a great many words. They are subtly combined with other concepts in expressions such as respond, counteract, interrupt, be vulnerable to, and force. There is a spectrum of concepts from "is associated with" to "is the direct cause of". The choice of an exact place on this spectrum is made more difficult because causality is used transitively and many degrees of abstraction are possible in an explanation. Representation of time is another difficult area. The links of figure 1 say nothing about this. They are written in a form that suggests that A first causes B which then causes C. In reality, the causality is continuous and constantly changing. A more powerful description is needed for describing such a smooth succession of states through time. Without it the different pieces of news are hard to relate in more



detail. These are only two important areas from many. Others are necessity, belief, chance, and comparison, and it is easy to find more. But even with a better theory of representation than we have, the information for producing a more complete description is not in the passage. It must be derived also from an extensive world model.

Natural language has many devices that increase its expressivity and much information is communicated implicitly. The interpreter is given the task of making this explicit. It is precisely because people can do this so easily, that a representation may appear to capture the meaning of a passage when, in fact, the person viewing it is filling in a lot of the gaps. The fact that a person can use the information in a representation does not imply that it is in a suitable form for a program. Nor does the fact that it somehow looks formal imply that all the information is represented adequately.

In particular, natural language facilitates abbreviation. Several referents in the example, such as "brokers", "selling" and "prices" are abbreviated, and the reader is expected to infer their full form. "In futures markets" sets a context that provides helpful information for completing these descriptions and deciding that "brokers" are probably futures market brokers, "selling" is selling of grain futures, and "prices" are grain futures prices. Unfortunately for the design of expert programs, the completion process is not easy to specify as a precise algorithm. Syntactic methods are not enough and, in many cases, familiarity with the domain is critical. To be confident that "prices" refers to grain prices, we use the knowledge that heavy selling of futures of some commodity tends to make the prices of that commodity lower. The nominal structures of the passage, such as "spreading dockworker strike problems", and "grain exporting terminals" are also abbreviations. The actual relationships between their component concepts can be complex and unusual. In the example, the strike is *by* the dockworkers, the problems are *due to* the strike, and it is unclear whether it is the problems or the strike that are spreading. Interesting work has been done on the problem of nominal expressions [Brachman], but a program that discovers correct relationships in a significant number of cases is not yet in sight.

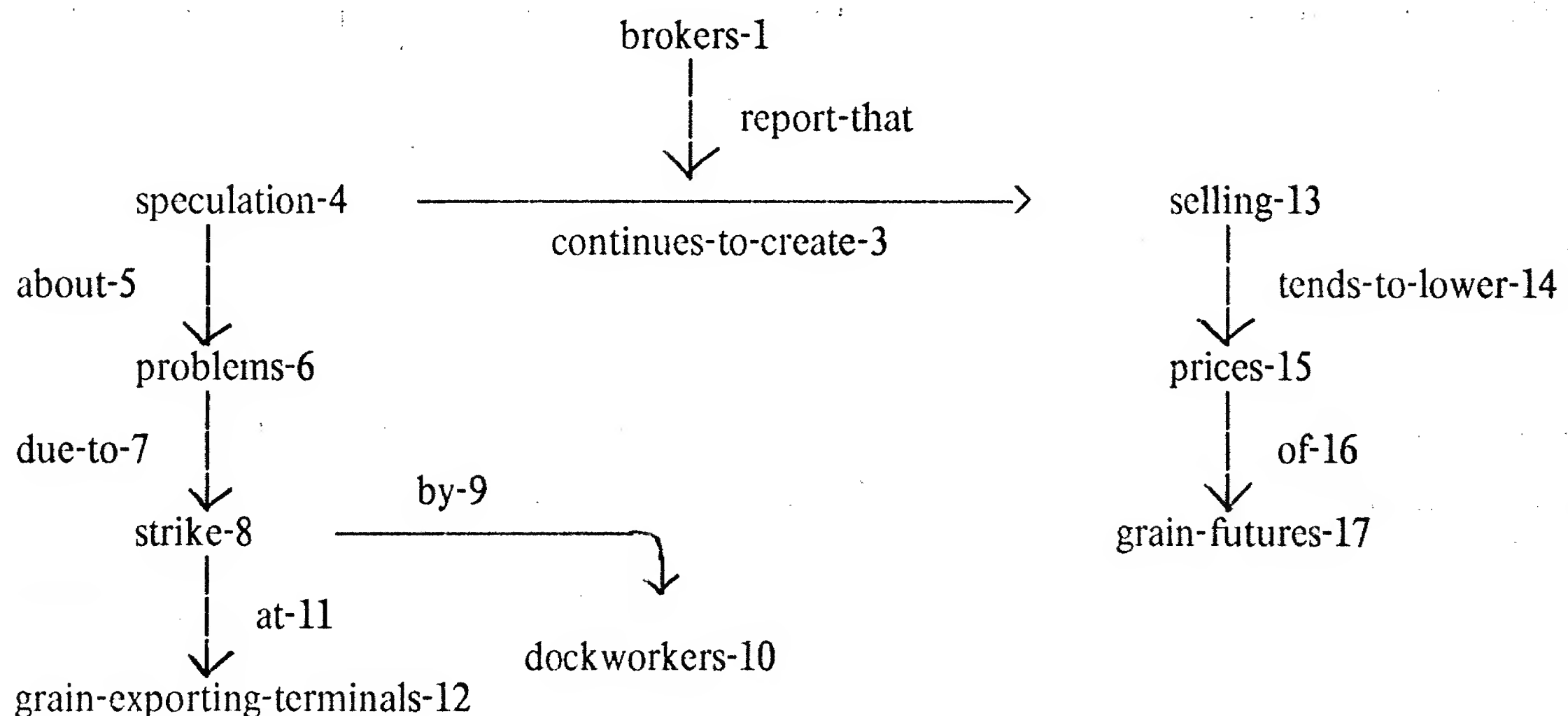


The passage could have been paraphrased in many ways while retaining a more or less constant content. If a program is to reason about it, either the paraphrases must be reduced to a canonical form or the similarity of the many forms must be recognizable. I cannot convince myself that a canonical form is the right way to proceed and, even if it is, I am unable to find an adequate one. My impression is that handling the various forms of equivalence, and the potential computational explosion associated with them, is a key problem of representation theory.

Figure 2 shows a more detailed representation than does figure 1. Although it is fairly typical, there are many questions about its interpretation. One interpretation of the diagram is as a simple parse tree with no features such as part of speech labels, tense markers, or determiners. In this case, the entire structure could be considered as a single factual statement and must be clearly delimited from the other statements in the data base although they may refer to the same entities. A scheme for doing this would partition the individual links of the data base. A contrasting interpretation is that each node or link of the structure represents an object or a relationship between objects. The two interpretations seem to suggest two levels of representation, one closer to English and the other a deeper representation. However, this distinction is unclear, because similar requirements must hold at each level. There must be a means of defining structures, a type of structure that represents an assertion, and a type that represents a description. The two interpretations of figure 2 simply

---

Fig. 2. A more detailed representation of the example.



reflect different decisions on grouping the parts of the figure into structures. It is important to be clear about the grouping that is intended. It is also important to indicate which structures are meaningful in themselves and which are only grammatical parts of larger structures that are meaningful. One of the major difficulties with using FRL was that no facilities are provided for handling these distinctions, and the usual interpretation of its extended property list structure ignores them.

There are different ways to assemble the parts of a structure. Suppose that *strike-8* represents a particular strike event. It is at the same time a single symbol and a complete referring structure. We should then interpret *by-9* and *at-11* as assertions about that strike. The force of this is that there is a strike, it is by dockworkers and it is at grain-exporting-ports. How then do we distinguish the sentence "the strike by the dockworkers is at grain exporting ports" from "the strike at grain exporting ports is by the dockworkers?" There must be two ways to attach a link to a node, either attributively or definitively. Woods discussed this distinction in his famous "what's in a link" paper [1975]. If we regard the link *by-9* as definitive, making the expression "the strike by dockworkers", that expression acts as a noun phrase description and refers to a particular real entity just as *strike-8* did. It follows that even in the underlying representation there must exist structures which act intensionally as descriptions. We should not take separate views that language deals only with descriptions, while deeper representations deal only with their referents. Even if there are two levels, they must both deal with expressions and use them for making assertions and referring. *Strike-8* is no closer to the real world than "the dockworkers strike" just because it is a generated node-name and the other is a sequence of words.

A referent-finding process can only discover some other description that refers to the reference, since the actual entity referred to is a phenomenon in the real world and is not directly accessible. The two descriptions can be related with a coreference link [Martin]. If the same entity is referred to in two passages, the referent-finding process must discover that the two expressions corefer. Referent-finding is a difficult process to implement, demanding the use of semantics and pragmatics. In many cases a referent can be found immediately, but further information is often needed. Indefinite expressions like "one of the cars belongs to Fred" are good examples. Even the

speaker may not know which car it is, so the disambiguation process may be left incomplete to be continued later. I believe referent finding is a fundamental component of reasoning.

The referring expressions that a commodity data base is likely to contain must not be interpreted as statements that the referents exist. For example, the presence of the *strike-8* symbol should not be taken as an assertion of the existence of some strike in the real world. While brokers reported speculation about a strike, they might have been mistaken. Neither a report of an event nor speculation about an event definitely imply the event occurred. In the reported world, however, the strike is supposed to exist.

These problems are not necessarily intractable, but there are a great many of them and they impede the progress toward an intelligent assistant. Before attempting to design an expert program that understands news articles, it would be wise to address some of the thorny representation issues. These conclusions led me to other approaches for the commodities assistant project and I describe them next.

### **The simulation approach**

An arbitrary collection of news articles does not readily support the development of an intelligent assistant for a real commodity analyst. The major difficulties are their unbounded character, the vague and fragmented nature of the data they provide, and the difficulty of representing natural language information well enough to cover them. These difficulties suggest it would be better to handle information about a more manageable area of the commodities market and to make certain that the data is complete enough to support a coherent representation of the behavior within that area. Also implied, is a model of the commodity market's underlying mechanism of sufficient detail that a program might reasonably hypothesize causal relationships to interconnect the data. It is important to include numeric information to support qualitative statements about increases, decreases and trends. This is because of the large part played by numeric variables such as stocks, prices, shipments, and production. Because of the size and complexity of the grain market, I believe another requirement is access to a commodities expert and to on-line commodities data.



My second approach evolved from these requirements. To reduce the domain to manageable proportions and to have expert understanding of it, I decided to use a much simplified simulation of the grain market as a test-bed [Stansfield]. Although this simulation would cover only a subset of the real world, within its range it would precisely define the behavior under consideration and the mechanism that was supposed to produce it. In other words, the domain would be contained and coherent.

I envisioned several uses for a simulation of this nature. Data generated by it would be fed into the analyst's assistant program which would produce descriptions, explanations, and forecasts. This would overcome the lack of real on-line data. The program would also use the simulation as a data structure, regarding it as a model of the many causal influences in the grain market. The task could then be viewed as discovering an instance of the simulation to fit given information. Hypotheses could be tested by running them on the simulations. To allow this, the program must be able to access the structure of the simulation.

I built a simulation language, SIMSYS, using the frame data structures of FRL. SIMSYS was based on industrial dynamics [Forrester] and its simulation language, DYNAMO, [Pugh]. I made modifications and extensions, but will not describe SIMSYS in this memo. Although, the use of FRL frame structures made simulations very easy to set up and access, they were large and slow.

Two simulations of the market were attempted. Since a large proportion of commodities reports concern weather conditions and their affect on crop production and on price, one simulation dealt with the effect of weather on corn production. In different stages of its growth, corn is affected differently by precipitation, temperature, and sunlight. Other important aspects are the top-soil moisture, the sub-soil moisture and the amount of fertilizer available. Frame structures were set up to represent these concepts and a simulated farmer decided when to plant and when to harvest. The other simulation modelled how the commodities market operates as a game between producers, consumers, speculators and merchants. These agents have a repertoire of actions such as buying, selling, transporting and storing, all of which affect the price. The strategies of these agents determine the particular actions they choose in any given circumstances.

Unfortunately, both simulations were far too simple. The grain market is so complex that to construct a simulation that supports an explanation of any realistic behavior is a full-time project for a commodity expert. Existing commodity models are predominantly econometric. They consist of small sets of equations whose parameters are decided on by multivariate analysis. Such models do not lend themselves to mechanistic explanations. The few simulations [Meadows] [Raulerson] that use a more mechanistic style of model have a very small repertoire of behavior types. Possibly, these would support a feasible AI program that would explain the output of such a simulation and would act as a question-answering interface between the simulation and a user. Such a project is ambitious because of the representation issues that arise in qualitatively describing continuous behavior. It would be important to restrict its scope to a simple, well-specified simulation.

The test-bed simulation approach quickly led away from the problem of designing intelligent assistants. It was soon clear that an analyst's model of the world is more flexible than a simulation. A commodity simulation powerful enough to model a significant segment of market behavior would create similar information management difficulties as does the real world. Simulations also require complete, consistent information within the area they model. Clearly, real analysts can summarize information to focus on important aspects, and they can also handle incomplete or false information.

These conclusions led to a third approach that addresses the intelligent assistant more directly. In the remainder of this memo, I describe this approach and try to clarify its problems.

### **Qualitative reasoning about graphs.**

The third approach comes directly from two premises. The first is that numeric information must be included. To ensure realism, it is important to use actual figures rather than to generate them using a simulation or similar means. Since no on-line figures were available to me, I gathered a usable set from government statistical reports. I collected about twenty tables of figures for corn, including stocks, prices, shipments, receipts, and exports at various locations. They were normalized to be weekly over a period of 18 months. At the same time, I gathered commentary that described events in the grain market during the sample period.

Table I. Types of features of graphs.

MEASURE	FORM
Absolute size	Peaks
Comparison	Troughs
Area	Trends
Gradient and slope	Sudden changes
Curvature and second derivative	
QUALITY	TEMPORAL RELATIONSHIPS
Smoothness	During
Variability	Before

---

A second constraint is that the model used by the program to construct explanations must be close enough to that of an expert to facilitate communication. I assume this means that the model is symbolic and that its symbols correspond roughly to what we think of as "concepts". I chose the paradigm of sets of rules expressed as statements in a logical language with relations, properties, connectives and quantifiers. Within such a rule-based approach, one can see how a program might explain its hypotheses [De Kleer].

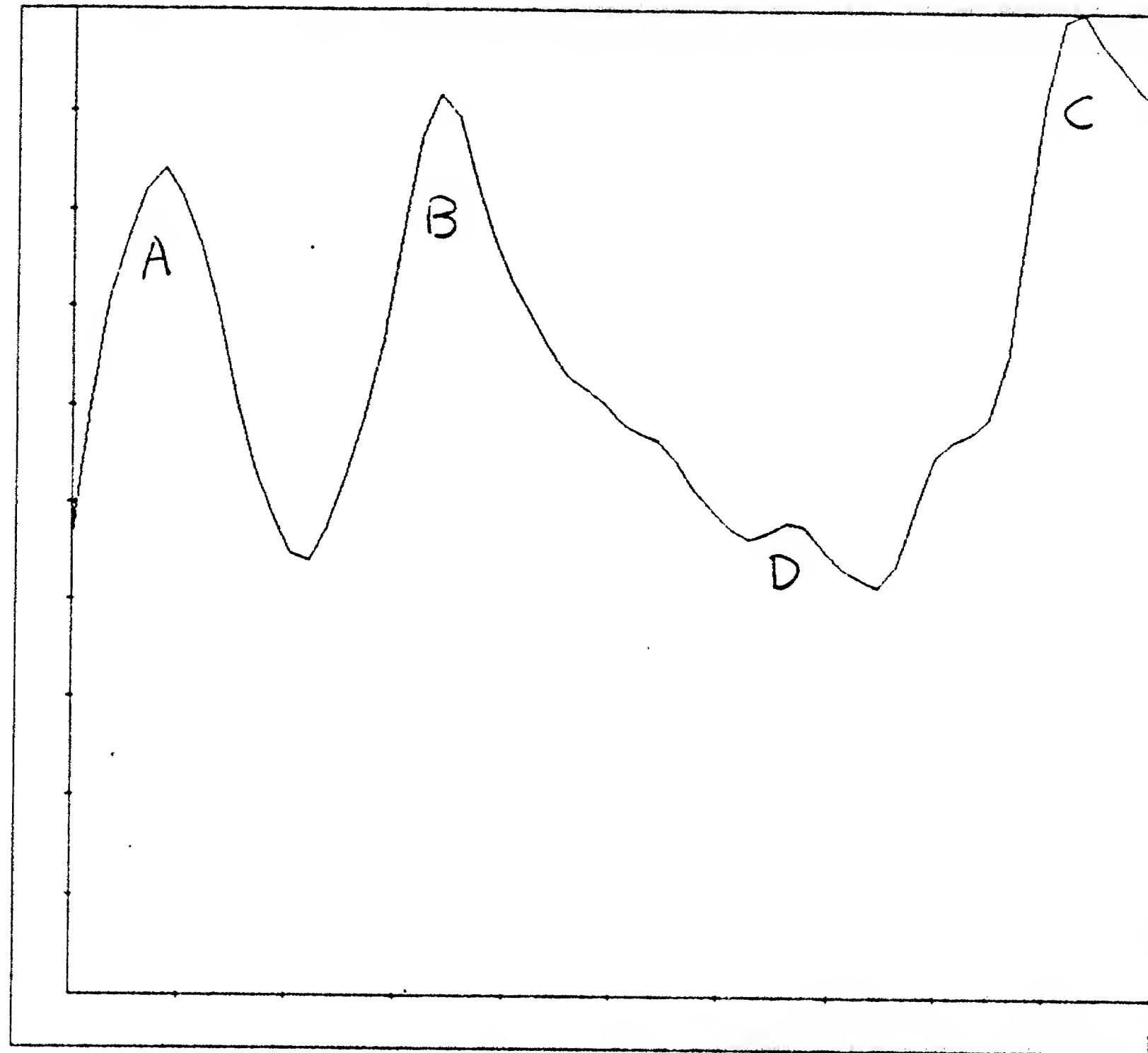
Qualitative reasoning about graphs is a means by which descriptive symbolic methods can be used to handle tables of numeric information. These methods are usually applied to commentary while more mathematical methods are applied to quantitative data. Examples of the quantitative approach are the equation-based econometric models, time-series analysis, and simulation.

Figure 3 demonstrates how a graphic representation of a table of figures vividly reveals a number of features of the data that are hidden in the table. The graph has a clear structure that can be described symbolically as a set of features and the geometric relationships between them. For example, at three places, A, B, and C, the dependent variable has a significant peak. The value is locally maximum at any peak, and the value is relatively high around it. It is easy to visually compare an ordinate at one location with another or with typical values for the graph. In this example, all three major peaks are of similar height and their maximum values are absolutely large. In contrast, the minor peak at D has a low absolute value even at its maximum point. A graph can



Fig. 3. Some barge basis figures and a corresponding graph.

NAME BARGE-BASIS  
FNAME GRAPH-25  
ITEM ITEM-31  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 39.529911  
METRIC 1.49254067



date	basis
09-10-76	14
09-17-76	26
09-24-76	40
10-01-76	31
10-08-76	31

date	basis
10-15-76	30
10-23-76	30
10-29-76	30
11-05-76	21
11-12-76	22

date	basis
11-19-76	19
11-26-76	18
12-03-76	15
12-10-76	19
12-17-76	23

date	basis
12-24-76	24
12-31-76	26
01-07-77	31
01-15-77	37
01-22-77	35

also be visually segmented into parts which have a constant property. There are several segments of this graph where its value increases steadily from left to right, and qualitative values can be put on the slope of these segments. Table 1 summarizes some of the important descriptive features of graphs.

I hypothesize that an expert who has to make general conclusions about a large amount of numeric data relies heavily on descriptions of the visual properties the data has when presented graphically. These descriptions conveniently summarize the data, and a long series of figures may be expressed as a few qualitative statements. By describing a graph, we also make its information available to symbolic reasoning methods. These properties led me to rephrase the problem situation as follows: develop a program that will take a set of graphs of significant variables in the grain market over a period of time, and will explain some of the events that happened in terms of the features of the graphs. Informal examples of such explanations can easily be found in magazines such as *Time*. Typically, graphs of several leading economic indicators are presented with a discourse that relies heavily on them, and makes frequent reference to their features. The features correspond to events and states of the economy.

The features of graphs are purely syntactic, so their explanation must depend on a knowledge of what the graphs represent. For example, figure 3 is a graph of barge basis plotted against time, and both variables represent real-world entities. The features of this graph depend directly on these entities and indirectly on the factors that affect them in turn. A program that explains graphic features must therefore have rules that specify how the relevant real entities relate. They will include rules describing how events and states influence or lead to others and may also include rules that state how events relate without specifying any cause. The rules are needed for discovering semantic relationships between the syntactic features of graphs. They must be general enough to apply to many situations. Also needed is a more static description of the commodity world including descriptions of the structure and function of entities such as ports, routes and markets. The set of general rules together with the static description is the conceptual world model of the expert.

Consider how an expert might proceed to explain features of a single graph. The first step is to describe the graph to find a set of features to be explained. Next, the graph's variables are examined to see where they fit into the world model. If the graph is a time series, the independent variable will be time, and the dependent variable will represent some other real world entity. There may be rules in the model that involve these variables and that apply to the features needing explanation. These rules are likely to involve other variables. The expert would see whether the data base contains graphs for the new variables and, if not, whether they can be calculated or approximated from others. Any new graphs that are found will also have their features described. At this point, the rules can be applied, possibly setting up plausible relationships between the features of the new graphs and those of the original. Some relationships will only be hypothesized, in which case ways may be suggested for verifying them. It is also likely for multiple explanations to be put forward. In summary, an attempt to explain features of one graph leads to a search through the data base for related graphs. Search paths are opened by rules that describe how variables relate. The rules will match when related graphs have appropriate features. Since events which explain others may themselves require explanation, the process can be repeated as far as desired.

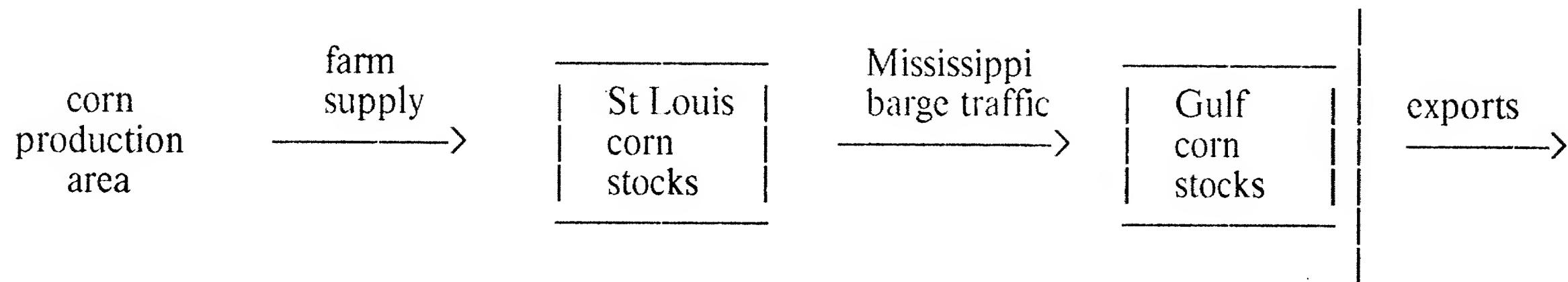
### **A simple scenario**

To see how this works, we will look at a simple scenario involving the flow of corn through part of the transport network of the United States. Figure 4 shows the situation schematically and includes the Mississippi river and two important points on it, St Louis and the Gulf Ports. Corn is produced in the corn belt and much of it is shipped to the Gulf for export and moves down the Mississippi. St Louis is one of the loading points and its price is therefore representative of an inland price. Variables that are important in this example are the prices, stocks, receipts and shipments at both locations.

First, consider the system in the simple equilibrium case. In this state, there is a straightforward short-term reaction to a change in one of the variables. An increase in exports can be interpreted as a stronger pull from the Gulf and therefore as an increase in the demand there. The Gulf price will begin to increase and the demand for corn for shipment from St Louis will also increase. By a similar argument, this will increase the price at St Louis and the demand from farms in the corn



Fig. 4. Diagram of corn flow to the Gulf from St Louis.

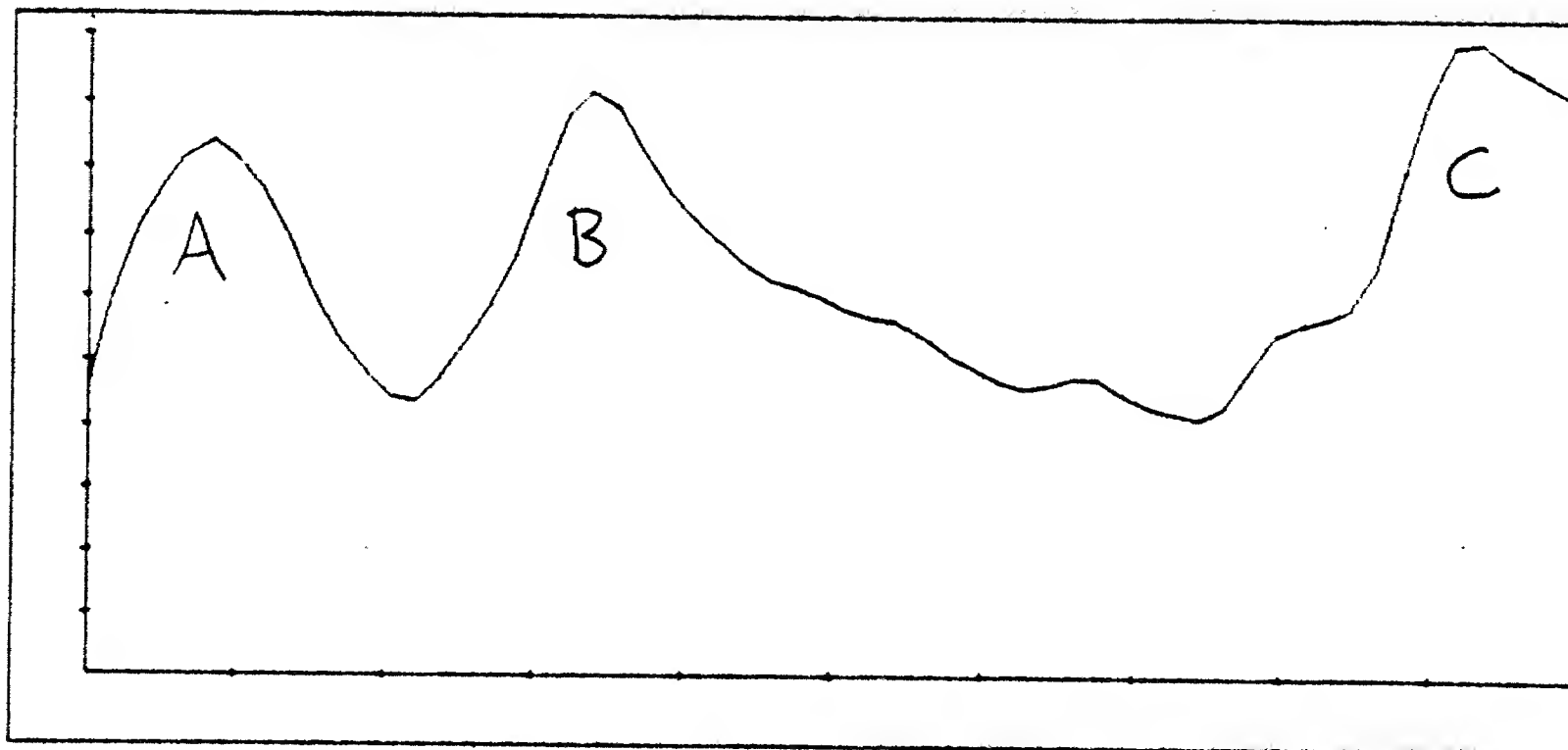


belt. The total short-term effect is for the change to propagate inland, so the causal connections are from right to left. If the change in exports had been a decrease, the effect would still flow inland but the increases and decreases would be reversed. If the initial disturbance had been in the corn belt, perhaps as an increase in supply due to harvest, the effect would have travelled out to the Gulf. Other types of event have different effects on the system. If the transport route is constricted and demand for transport is more than can be conveyed, there will be a shortage at the Gulf and the price will rise. This rise might not be propagated inland because there is no point in traders at the Gulf buying corn from St Louis that cannot be transported. However, if the restriction on Mississippi transport is large enough, the demand at St Louis will be reduced and the price there will fall.

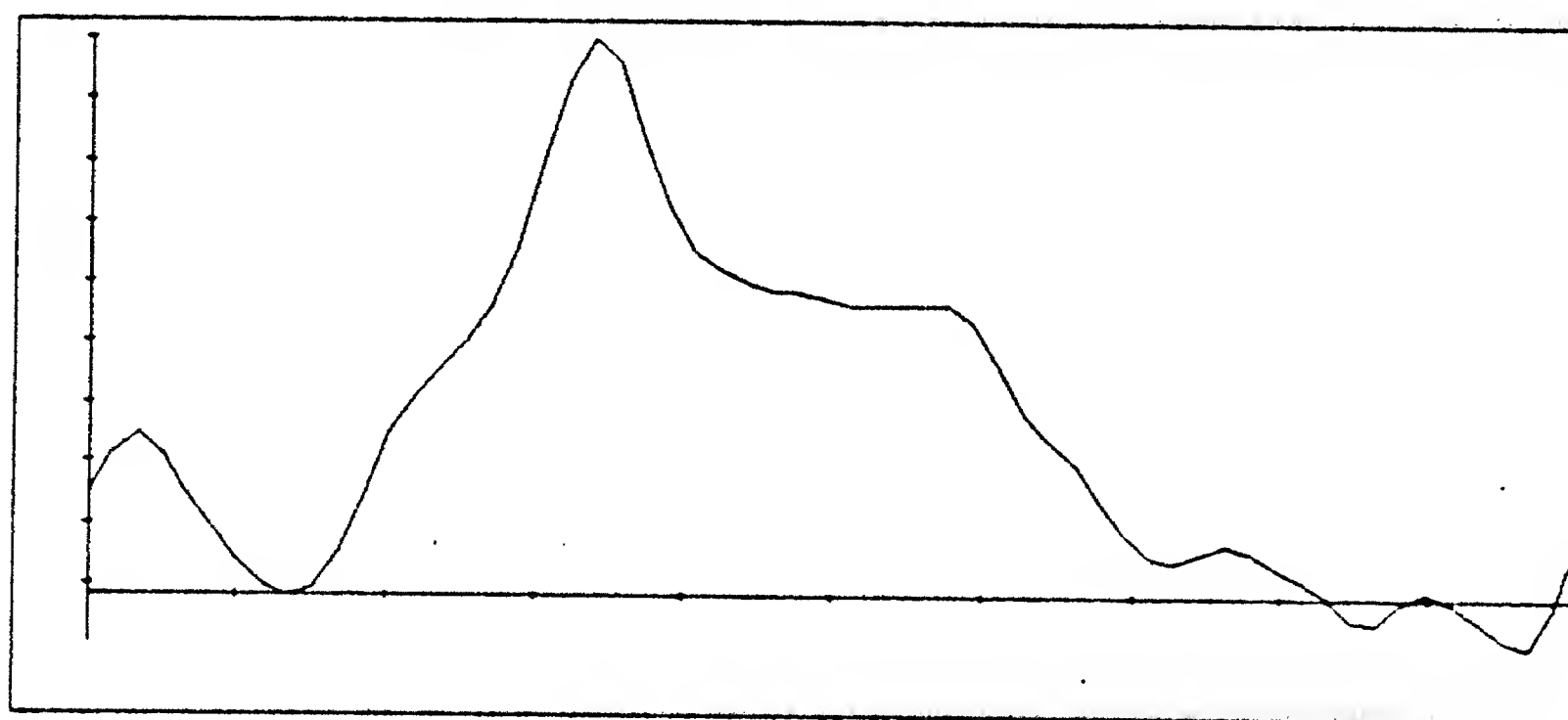
Figure 5 shows some real data about just such a situation. It turns out to be very hard to find real situations that can be explained as crisply as the simple case just outlined. I will give the best example I have been able to find and then describe some reasons for the scarcity. The top graph in figure 5 is the graph of barge basis from figure 3, and the example task is to explain its three major peaks. The program would know that the barge basis is the difference of the price at the Gulf and the price inland. This would lead it to examine graphs of the two prices involved. These graphs are also shown in figure 5. It is clear from them that explanations for peaks A and C will be different from the explanation for peak B. The peaks at A and C were the result of very low St Louis prices and somewhat low Gulf prices. The program must set up subgoals to account for these conditions and their explanation requires background knowledge. Since each low occurred around the end of a summer, it can be hypothesized that harvest was in progress and that the low inland prices were the result of large farm supplies. In contrast, the peak at B was the result of high prices at the Gulf.

Fig. 5. Barge basis, St Louis Price, Gulf Price.

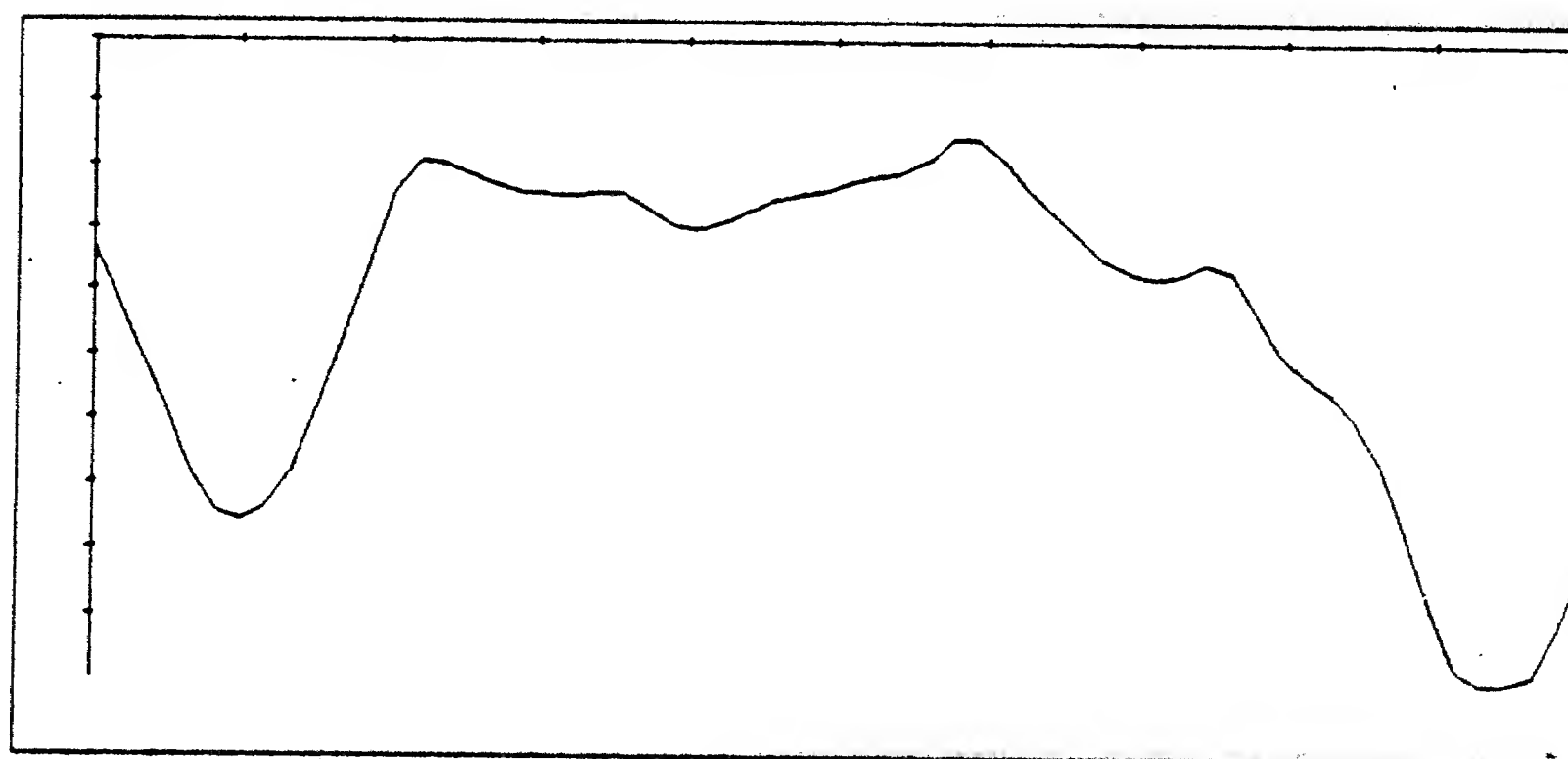
NAME BARGE-BASIS  
FNAME GRAPH-25  
ITEM ITEM-29  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 39.529911  
METRIC 0.6716433



NAME GULF-BASIS  
FNAME GRAPH-21  
ITEM ITEM-28  
XMIN 1.0  
XMAX 60.0  
YMIN -2.35586166  
YMAX 26.4287643  
METRIC 0.92236738



NAME ST-LOUIS-BASIS  
FNAME GRAPH-22  
ITEM ITEM-27  
XMIN 1.0  
XMAX 60.0  
YMIN -40.567127  
YMAX 0.0  
METRIC 0.654470794



There are two reasons why prices can be high: either low supply or high demand. Since supply and demand are intangibles, no graphs are available, but the program should know that receipts are an indicator of supply, and shipments an indicator of demand. At the Gulf, receipts are in close agreement with the total barge shipments from all points along the Mississippi, and shipments from the Gulf are mainly exports. Suppose the program has graphs for these variables as shown in figure 6. It could then tell that during the high at B, exports were still high. Barge receipts, however, were very low. Supply is short of demand and it is a reasonable hypothesis that this condition explains the data.

It is all very well to describe the scenario as I have just done, with a moderate amount of hand-waving, and with faith that the details can all be worked out and that a better program will deal with lots of examples. Unfortunately, this scenario was one of the few I could find where a reasonably clear explanation was evident, where the features matched up more or less exactly, and where the number of possible events and influences did not grow unmanageably. The difficulties are of several kinds. One problem is that it is very hard to handle rules in this domain. This is partly because statements of the rules bring up representation problems but also because it is unclear exactly what the rules should say. Other problems concern graph description and include problems of extracting and representing features as well as of deciding which features the program should even consider. Finally, there were resource limitations. Without an expert and with no on-line data, I found myself continually working on support issues that were time consuming and that led away from the major goal.

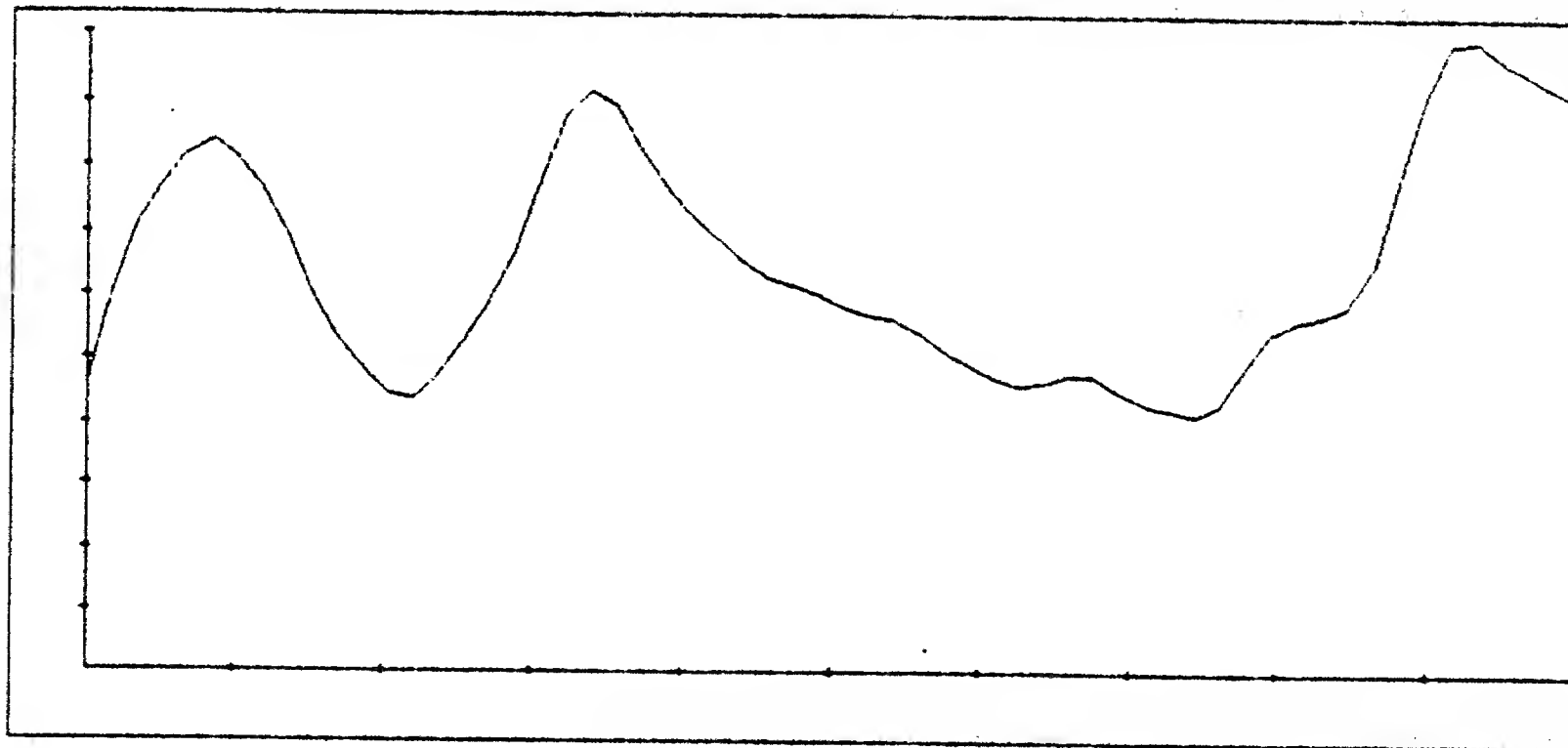
### **Problems with discovering and specifying rules.**

One of the ideas I had hoped to demonstrate in this research was the society of experts concept [Minsky]. The program's model of the grain market would be composed of many parts and each part would be expert in a single topic. Minor experts could be clustered together into a more generally applicable expert and would somehow combine their expertise by interacting. One of these, a supply-demand expert, would be an important component of the model. In the example, we saw how a crude supply-demand law could be used to explain a situation in which the price at the Gulf increased. I shall use supply-demand to demonstrate that rules for the grain market are

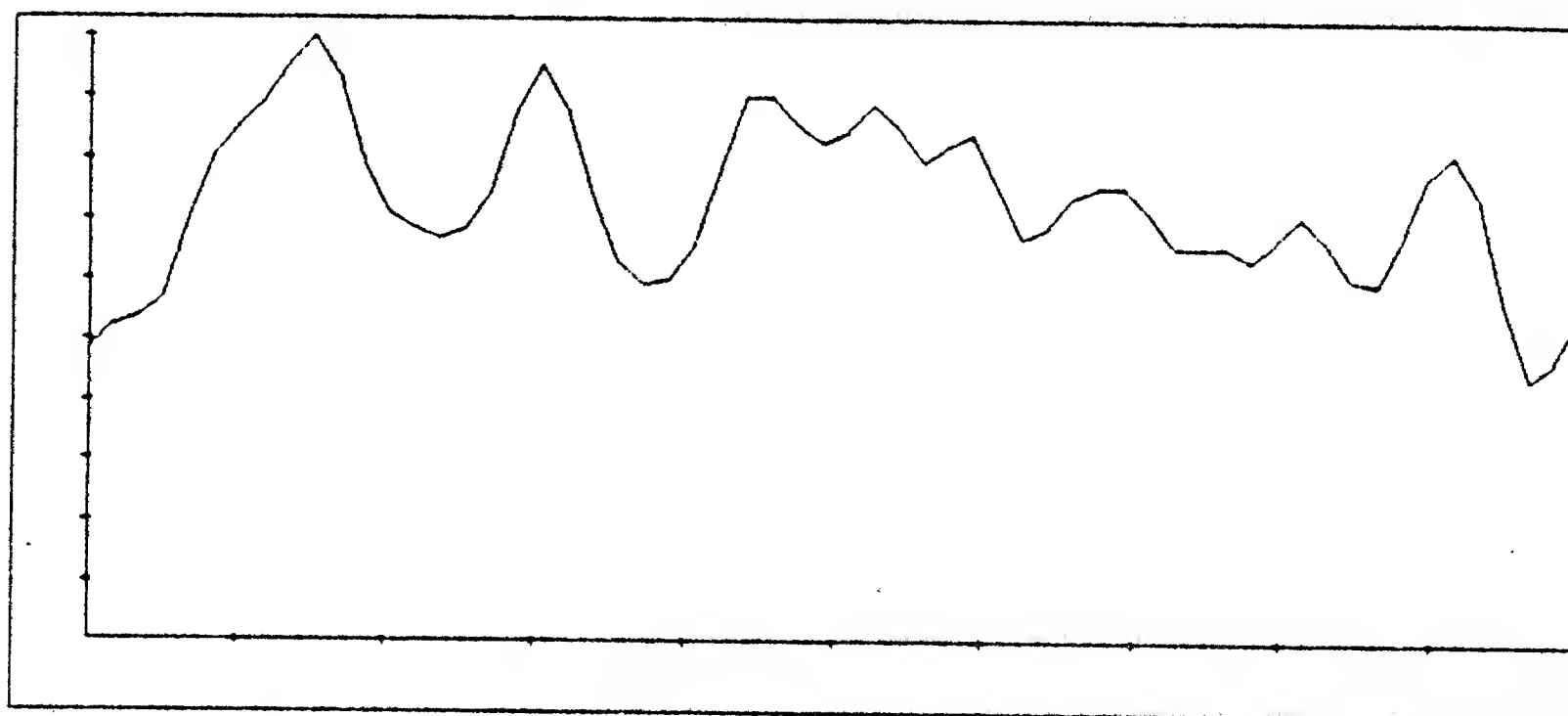


Fig. 6. Barge basis, Gulf Exports, Barge shipments to the Gulf.

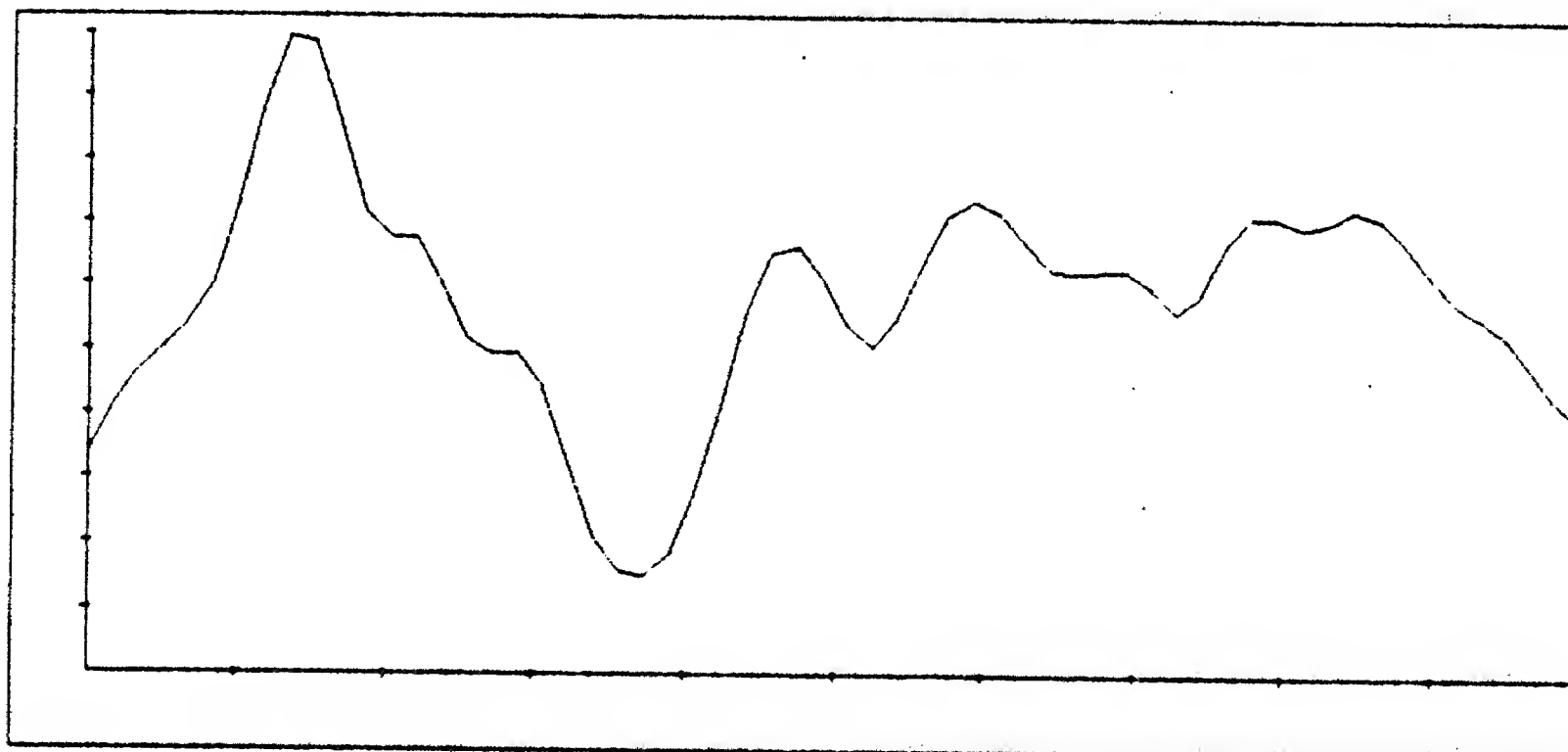
NAME BARGE-BASIS  
FNAME GRAPH-25  
ITEM ITEM-32  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 39.529911  
METRIC 0.6716433



NAME GULF-EXPORTS  
FNAME GRAPH-15  
ITEM ITEM-31  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 28318.0269  
METRIC 9.3756533E-4



NAME TOTAL-BARGES  
FNAME GRAPH-2  
ITEM ITEM-30  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 27562.0789  
METRIC 9.6328002E-4



complex and elusive. By probing into its structure I will show how a description of a supply-demand expert becomes progressively deeper. Much of this depth would be needed in an expert program for commodities, and this raises questions about the practicality of such a project.

In the equilibrium situation described earlier, the direction of change of price was easy to specify given the direction of change of supply or demand.

Supply increase	--->	Price decrease	--->	Demand increase
Supply decrease	--->	Price increase	--->	Demand decrease
Demand increase	--->	Price increase	--->	Supply increase
Demand decrease	--->	Price decrease	--->	Supply decrease

These rules can also be used in reverse to explain a price increase or decrease. As it turns out, each has two possible causes. This type of rule has been used in the context of simple electronic circuits in a program that determines the direction of change of voltages and currents for a given input change [De Kleer].

Unfortunately, these simple rules only describe what happens immediately following an equilibrium state. The grain market is usually out of equilibrium and dynamic rules are a more complex matter than static ones. Features of graphs extend over finite time periods and rules that can explain them must say something about dynamics. Suppose that we have a graph in which a price is falling. It might be falling because the demand is falling but it could also be that the demand fell significantly a little while back and the market is still coming into equilibrium. In other words there will not be a simple correlation between the features of the demand graph and those of the price graph. The relationship depends on such factors as elasticity of supply and demand. In order to adequately specify qualitative rules for supply-demand we must include something about these factors.

Further complexity arises because several influences can be acting at once. An increase in demand may be entirely offset by an increase in supply so that the price remains the same. If we could categorize all possible situations into a few prototypes, it would be easier to specify a rule but there are many intermediate cases.

The complexity does not stop here. In reality, the Gulf receives corn supplies from several places by means of several supply routes. The interaction of supplies from two different sources plays havoc with the supply-demand rule. It seems that simplifying the rules to manageable proportions leaves out information vital for explaining real data.

Supply, demand, and price are strongly interlinked, and the three variables are coupled by feedback loops with time delays. An explanation of the major features of their graphs must include these effects somewhere. All of these continuous effects place stringent demands on the form of the rules that an expert is supposed to have. One could always resort to econometric methods and use equations and regression analysis, but this would say little about how a human expert can deal with a complex supply-demand situation. In any case, such methods are inadequate for non-numeric information and for incomplete information.

I have mentioned only a few components of the supply-demand relationship and shown that it is difficult to describe the resulting behavior. The describable regularities might even be epiphenomena and not be explainable by causal rules at all. There may be no simple cause for a particular peak in prices, for example. If we examine the grain market for an adequate causal mechanism we move into a new area of complexity. Consider again the effect of an increase in demand for corn at the Gulf. The reaction of storage operators will depend on the amount of available storage, the orders that have already been placed for grain and when these are due, the availability and scheduling of shipping, and on many other factors. It is particularly difficult to handle expectations. Perhaps the storage operators think the increase in demand is temporary, or perhaps they had already anticipated it and acted accordingly. There are very many individual actors in the grain market, and its behavior is a combination of all their actions and strategies in a large and complex geographical arena. An expert must reason in terms of aggregates and fuzzy statements which makes it difficult to specify precise rules for the domain.

I believe my research effort would have been assisted greatly if I had been able to consult a commodities expert, however I think that the project would then have assumed a different goal. Rather than trying to design a program to assist an expert, it might be a more reasonable project to examine what an expert really does and some of the knowledge needed for this.

## Problems with feature extraction.

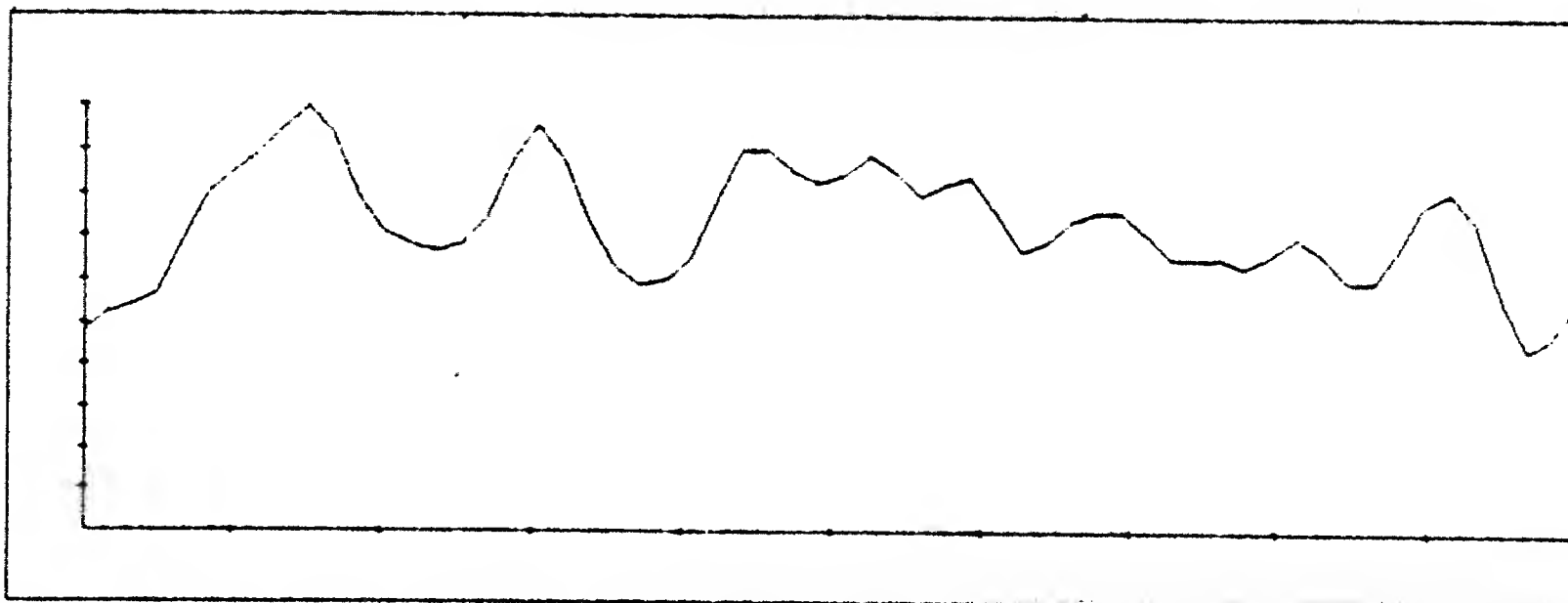
Graphical reasoning rules use representations of the features of graphic data as their raw material. To obtain these, a graphical reasoning system must include a component that can extract descriptions of relevant features from graphs. In this section, I describe a syntactic approach to feature description and discuss some of its shortcomings. The approach takes a reductionist view of graph description and assumes that there are primitive features of graphs from which more complex descriptions can be built. To take an analogy with speech-processing, the primitives would correspond to phonemes, while from an image-processing point of view they would correspond to lines and regions. However, the structure of a graph is not as clearly defined as the structure of a sentence or of a visual object. While it is relatively easy to define many kinds of syntactic feature, there seems to be no simple reason for using one kind as a primitive rather than another. As a result of my work on this, I believe that the decision can only be made for a particular domain, and only after research is done to discover exactly what an expert requires to support his arguments. For example, the problem of describing electrocardiograms [Stockman, Kanal & Kyle] is very clearly specified since the features useful to an expert are known beforehand. In the commodity assistant project it was not possible to examine what features a real expert would find relevant. Because of this, the feature extraction process could not be developed past a syntactic level.

The primitive features were taken to be segments of a graph over which some given property was constant. An example of such a property is that the gradient be positive everywhere in the segment. This criterion produces segments of a graph whose value increases steadily from left to right. Non-primitive features are compounds of these primitive segments. For example, a peak is a segment of positive gradient followed by one that has negative gradient. The gradient of a graph can be partitioned into more than these two intervals. A five-point qualitative scale would divide the gradient into high positive values, medium positive values, low values, medium negative values, and low negative values. A graph could then be segmented according to its gradient at any point. Segments could then be qualified with properties such as steep, or rapidly increasing.

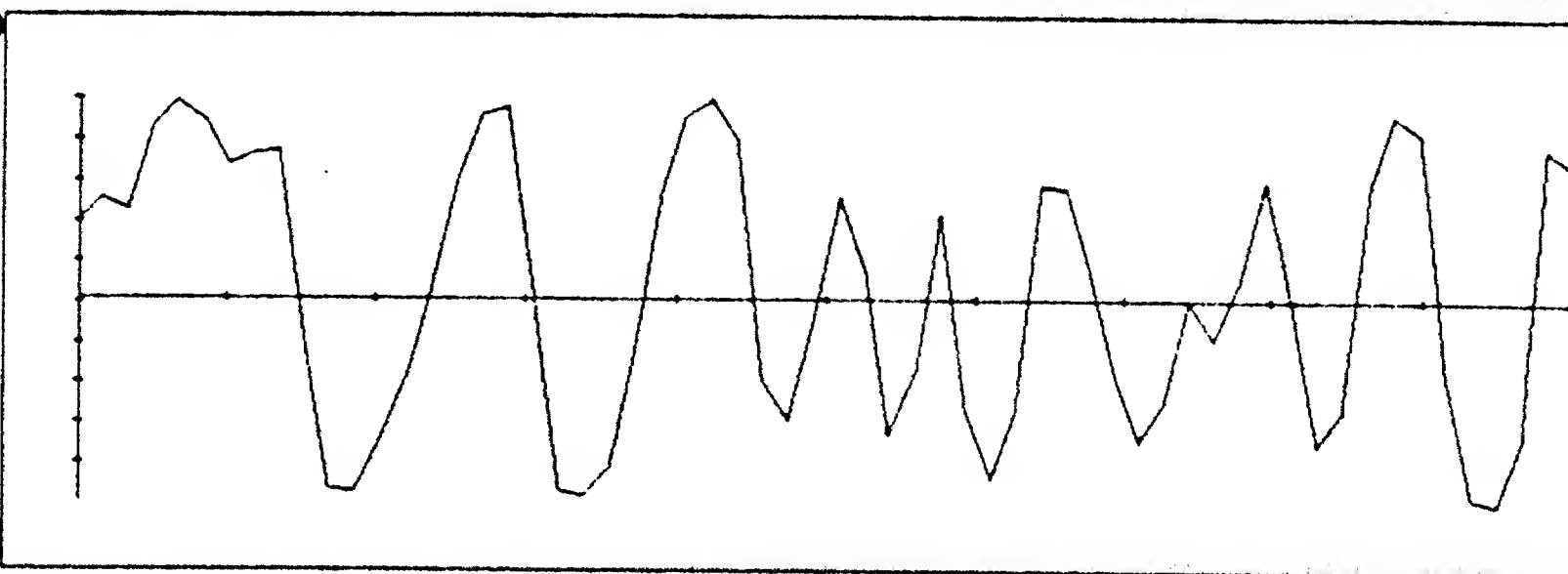


Fig. 7. Some functions of a graph.

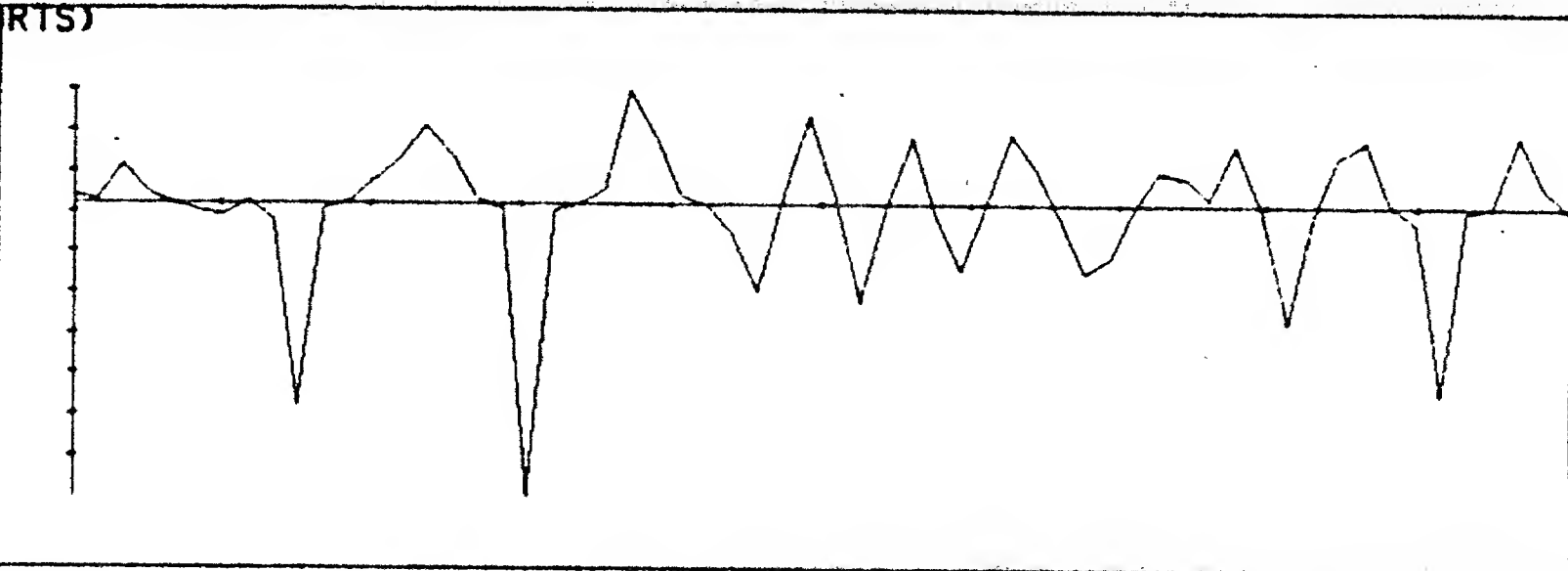
NAME GULF-EXPORTS  
FNAME GRAPH-15  
ITEM ITEM-43  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 28318.0269  
METRIC 6.2504355E-4



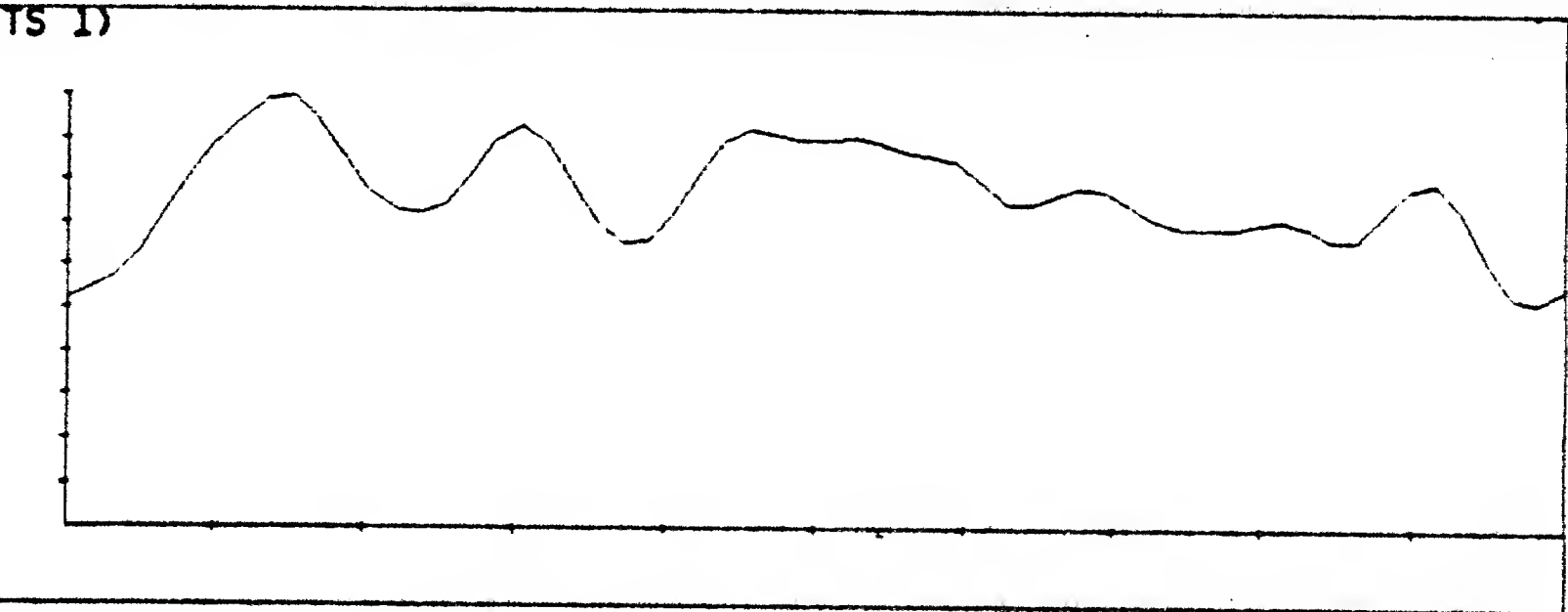
NAME (SLOPE GULF-EXPORTS)  
FNAME GRAPH-36  
ITEM ITEM-42  
XMIN 1.0  
XMAX 60.0  
YMIN -1.32402782  
YMAX 1.27202022  
METRIC 6.81805557



NAME (CURVATURE GULF-EXPORTS)  
FNAME GRAPH-39  
ITEM ITEM-41  
XMIN 1.0  
XMAX 60.0  
YMIN -2.8486265  
YMAX 1.10175821  
METRIC 4.4805763



NAME (SMOOTHED GULF-EXPORTS I)  
FNAME GRAPH-34  
ITEM ITEM-40  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 26874.512  
METRIC 6.5861661E-4



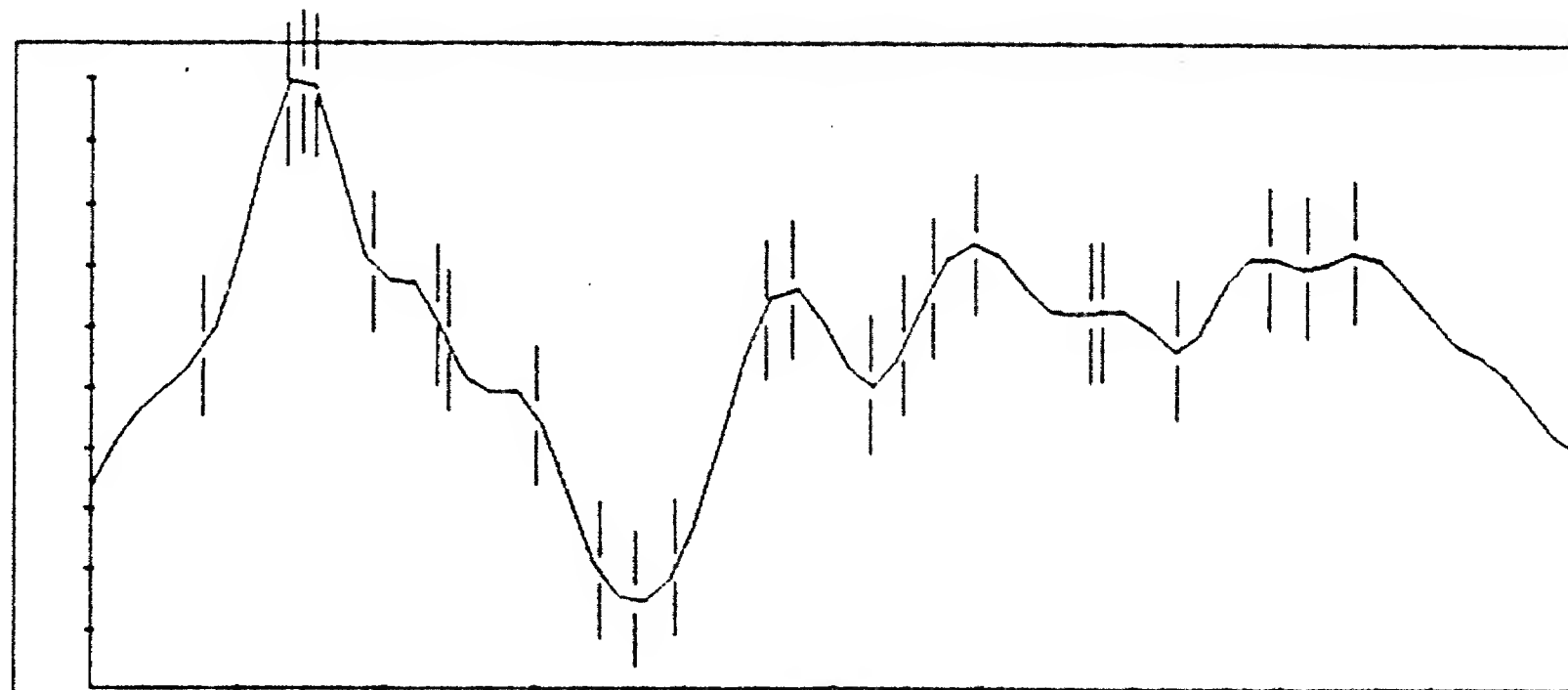
Besides the gradient, other functions of a graph can be used to segment it and my graph system had a repertoire of these. Typical functions of the graph of Gulf exports are graphed in figure 7. Some useful functions of a graph are its value, its second derivative, its slope and its curvature. Choosing five ranges for the value yields segments of high value, low value and so on. Segmenting according to the second derivative picks out segments where there is rapid change in the rate of increase or decrease, or where the gradient is fairly constant. The slope and curvature are a little more complicated since they are dimensionless quantities and their value depends on the relative scales of the axes in any particular drawing of a graph. If we change the scale of one axis, the graph will have different slope and curvature. For example, by squashing the y-axis, the graph will be compacted toward the x-axis and will show very little curvature. I assumed an expert would choose a particular scale for drawing any particular graph. At this scale, an average increase per unit time would be comparable in absolute length to the unit on the time axis. A graph with this scaling would show typical events clearly. To implement scaling, I associated a metric with each graph that specified its preferred scaling by relating the scales of the two axes. The metric then allowed the slope and curvature of graphs to be calculated.

Figure 8 illustrates how a graph can be segmented using one of these functions. It shows a graph of total barge shipments to the gulf that has been divided by gradient into twenty-four segments. The lower part of figure 8 shows how the values of the gradient graph fall into various ranges of y-value. The upper part displays the resulting segments of the original graph. Information about each segment is stored in an FRL frame and includes assertions that say what graph the segment is part of, what the end points of the segment are, whether the slope is large or small within the segment, and whether the segment is an increasing or a decreasing one. The frame structures in figure 9 show that a large number of assertions are generated by the feature extraction process.

Segmenting in this way results in some arbitrariness. For example, the upward slope at point A in figure 8 has three segments. The segment of high gradient is quite small and could easily have been smaller. It might be more appropriate to ignore it and to describe the entire upward slope as a single segment. Similarly, we might consider the slight dip at B insignificant. The region might be described better as a single large hump. The segment CD could have several descriptions. Although

Fig. 8. A graph segmented by gradient value.

NAME TOTAL-BARGES  
FNAME GRAPH-2  
ITEM ITEM-52  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 27562.0789  
METRIC 9.6328002E-4



NAME (GRADIENT TOTAL-BARGES)  
FNAME GRAPH-26  
ITEM ITEM-53  
XMIN 1.0  
XMAX 60.0  
YMIN -3753.31543  
YMAX 3906.8576  
METRIC 3.46597913E-3

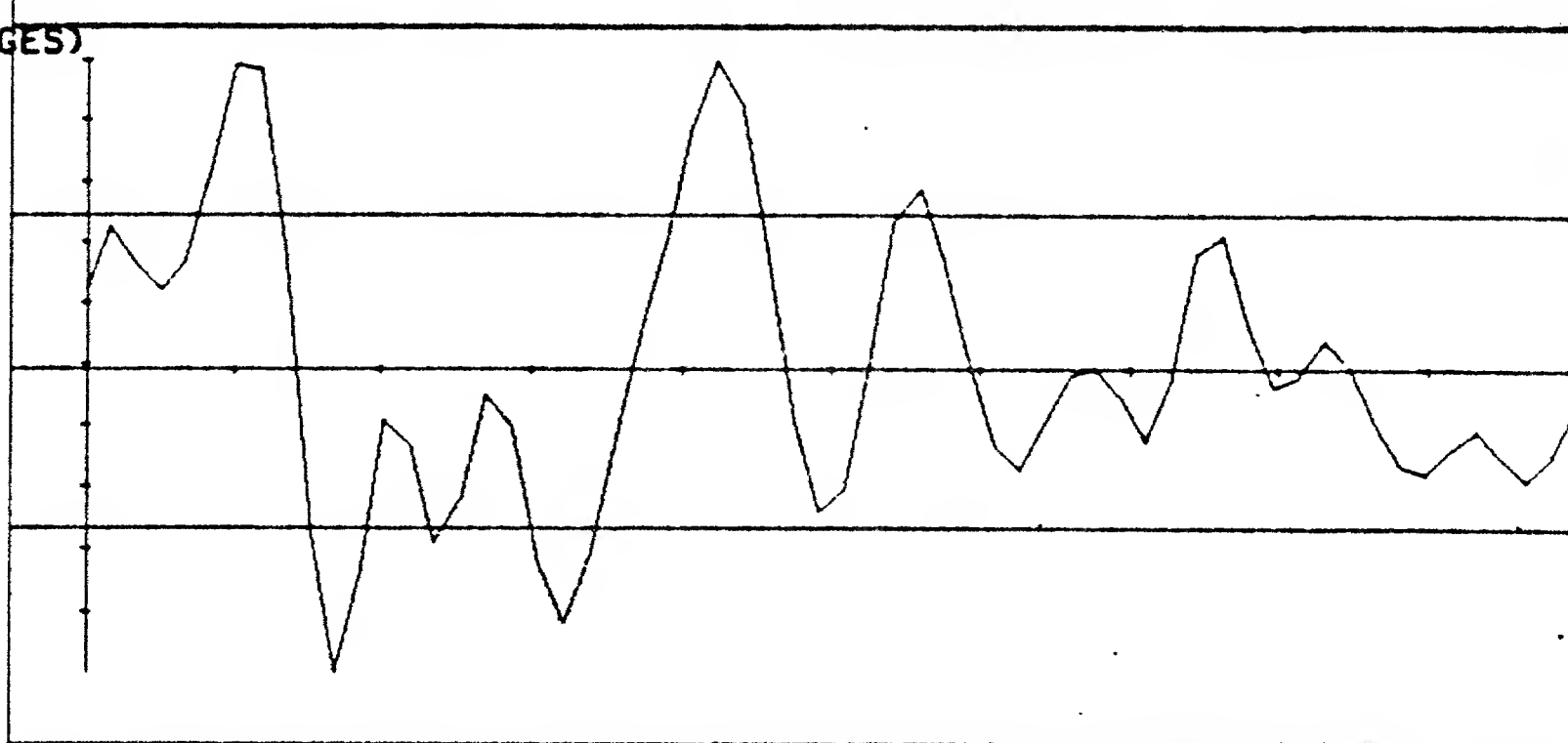


Fig. 9. Some frame structures for a segmentation.

(SEGMENTATION-54

(AKO	(\$VALUE	(SEGMENTATION)))
(OF-GRAPH	(\$VALUE	(GRAPH-2)))
(HAS-INTERVALS	(\$VALUE	((INTERVAL-51 INTERVAL-50 ..... )))
(HAS-SEGMENTS	(\$VALUE	((SEGMENT-55 SEGMENT-56 ..... )))

(SEGMENT-78

(AKO	(\$VALUE	(SEGMENT)))
(SEGMENT-OF	(\$VALUE	(SEGMENTATION-54)))
(HAS-INTERVAL	(\$VALUE	(INTERVAL-28)))

(INTERVAL-28

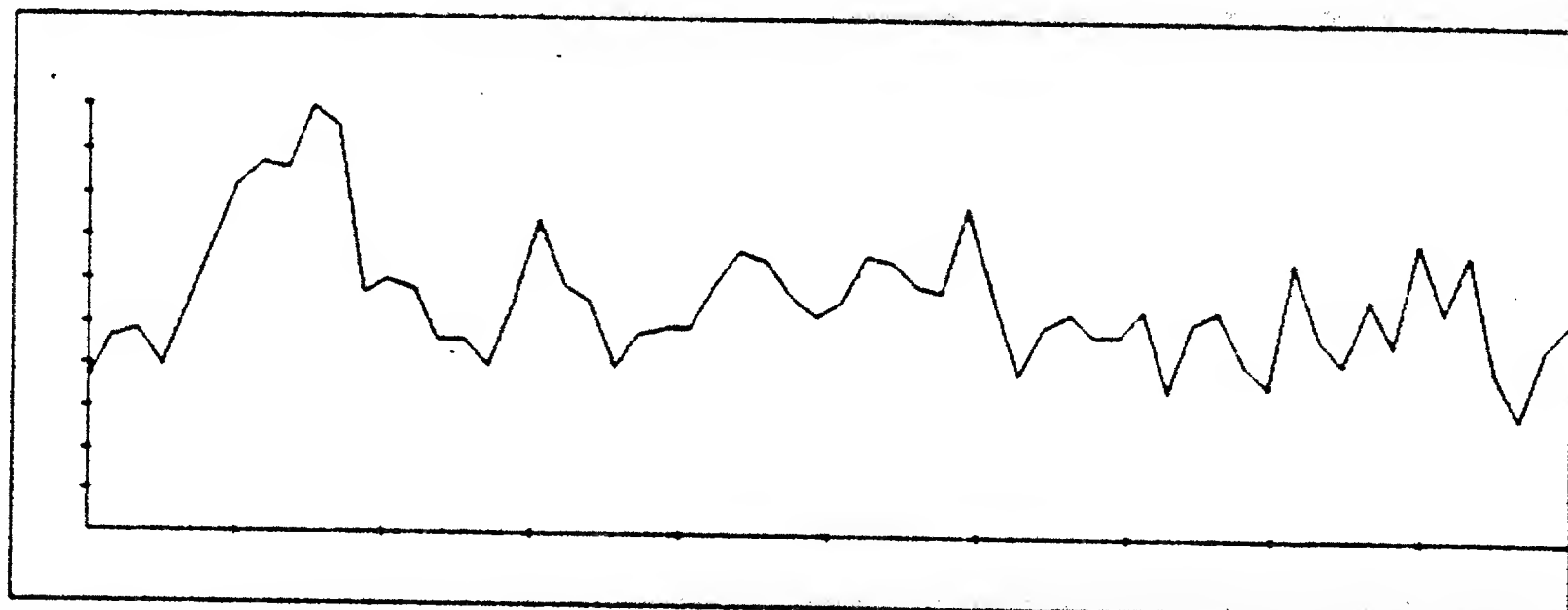
(AKO	(\$VALUE	(INTERVAL)))
(HAS-MIN	(\$VALUE	(1.0)))
(HAS-MAX	(\$VALUE	(5.48556495)))

its gradient is everywhere negative and it could be described as a single downward trend, it almost flattens out at E and F. The general problem in these examples is that the segmentation process uniformly applies a local procedure to the entire graph. Ranges are decided on for the whole graph and are applied point by point. To describe features of graphs as an expert does, we need a more global and more context-sensitive approach.

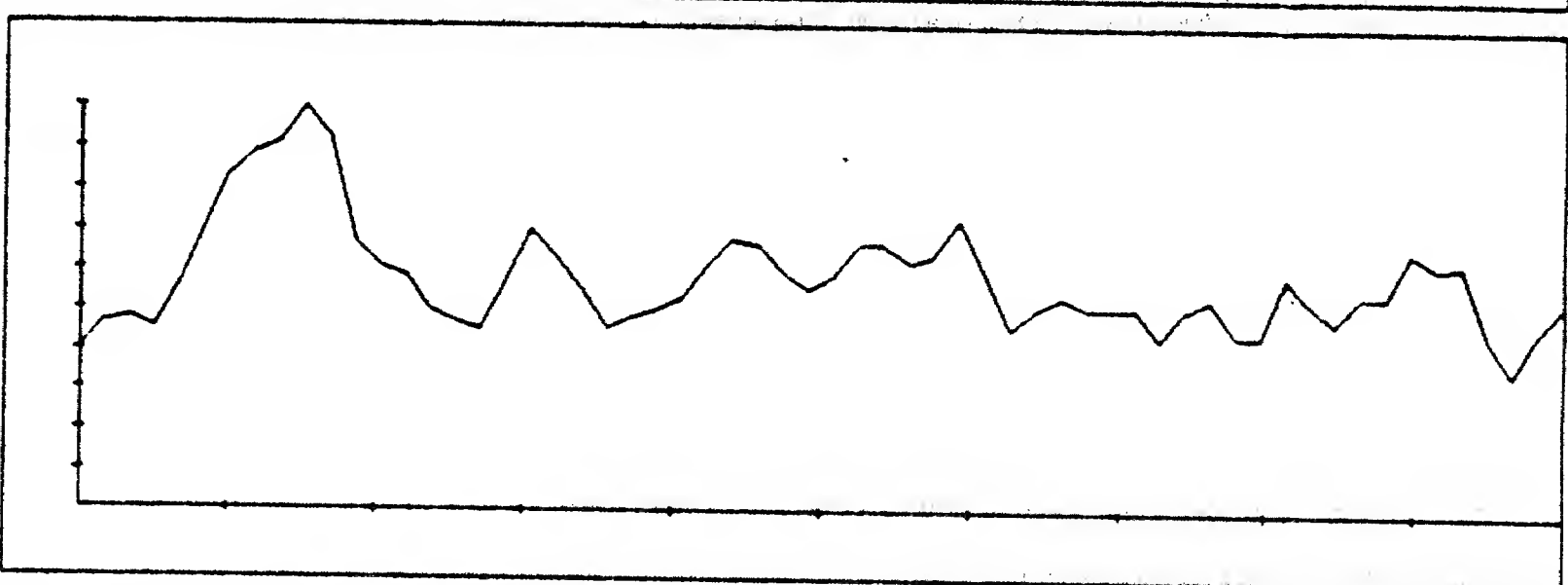
Part of the problem is to discover large-scale features despite the interference of small-scale local perturbations and noise. An attractive approach is to use smoothing. Figure 10 shows a graph of some original data together with two smoothed versions. The highly smoothed graph reveals the

Fig. 10. Different degrees of smoothing of data.

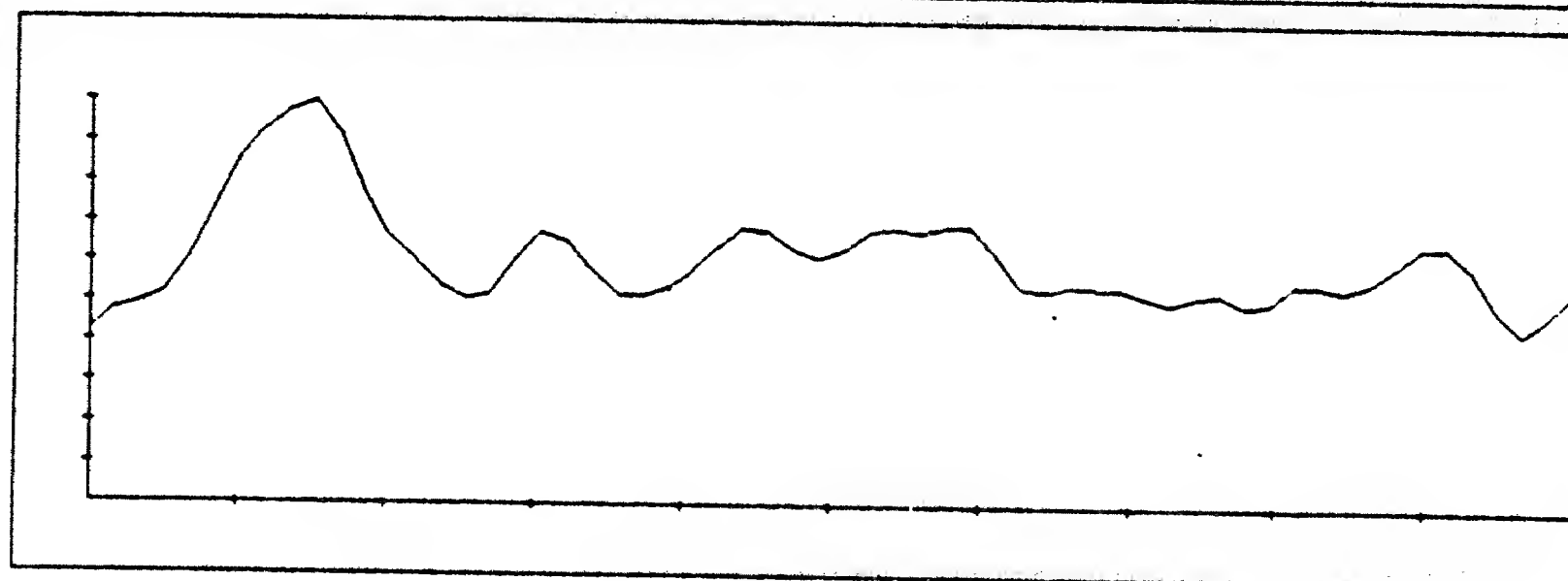
NAME DATA  
FNAME GRAPH-44  
ITEM ITEM-45  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 55800.0  
METRIC 3.17284303E-4



NAME (SMOOTHED DATA 0.5)  
FNAME GRAPH-46  
ITEM ITEM-47  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 53920.183  
METRIC 3.28262982E-4



NAME (SMOOTHED DATA 1.0)  
FNAME GRAPH-48  
ITEM ITEM-49  
XMIN 1.0  
XMAX 60.0  
YMIN 0.0  
YMAX 51123.9883  
METRIC 3.46217123E-4





broader features, while the less smoothed version still shows the more rapidly changing components. The many short-term features and sudden changes of the original graph interrupt the trends and obscure the relationships between this graph and others. From a semantic point of view, the long-term effects of major events in the grain market are overlaid with many shorter-term fluctuations. When the graph of figure 8 is smoothed enough, CD becomes a single steep downward slope and the double hump at B becomes a single hump.

Unfortunately, smoothing a graph results in an information loss. While smoothing does highlight large scale features, the location of their boundaries is obscured. Smaller scale features are still important as long as they are seen within their larger-scale context. What I hoped for was a series of progressively more abstract descriptions of a graph. The high levels of abstraction would describe only major features and the lower levels would fill out the details. I investigated the possibility of smoothing a graph by various amounts to extract features at different scales. At each level of smoothing the graph would be segmented and its features described. These descriptions would include progressively more features and be combined so that details were scoped within the context of larger features.

The smoothing filter I used came from the theory of edge-detection developed for image processing [Marr & Hildreth]. In their methods, an image is convolved with second-derivative gaussian masks of various degree. Convolution is effectively a weighted averaging process where the weights are given by a masking function. When used as a mask, the gaussian function acts as a smoothing filter. By considering the fourier transform of the gaussian function, it can be shown that it is optimally balanced between localization in normal space and in transform space. In other words, it will filter out high frequency components such as sharp perturbations but, at the same time, will not highlight periodic features. This is an advantageous property for an image-processing filter because visual objects are localized in space. By using the second derivative of these gaussians, sudden changes in intensity that are associated with the edges of objects show up as zero-values in the filtered image. The combined results of several sizes of filter are used in edge-detection. Large amounts of smoothing pick out the broader objects in the image and smaller amounts pick out the details.

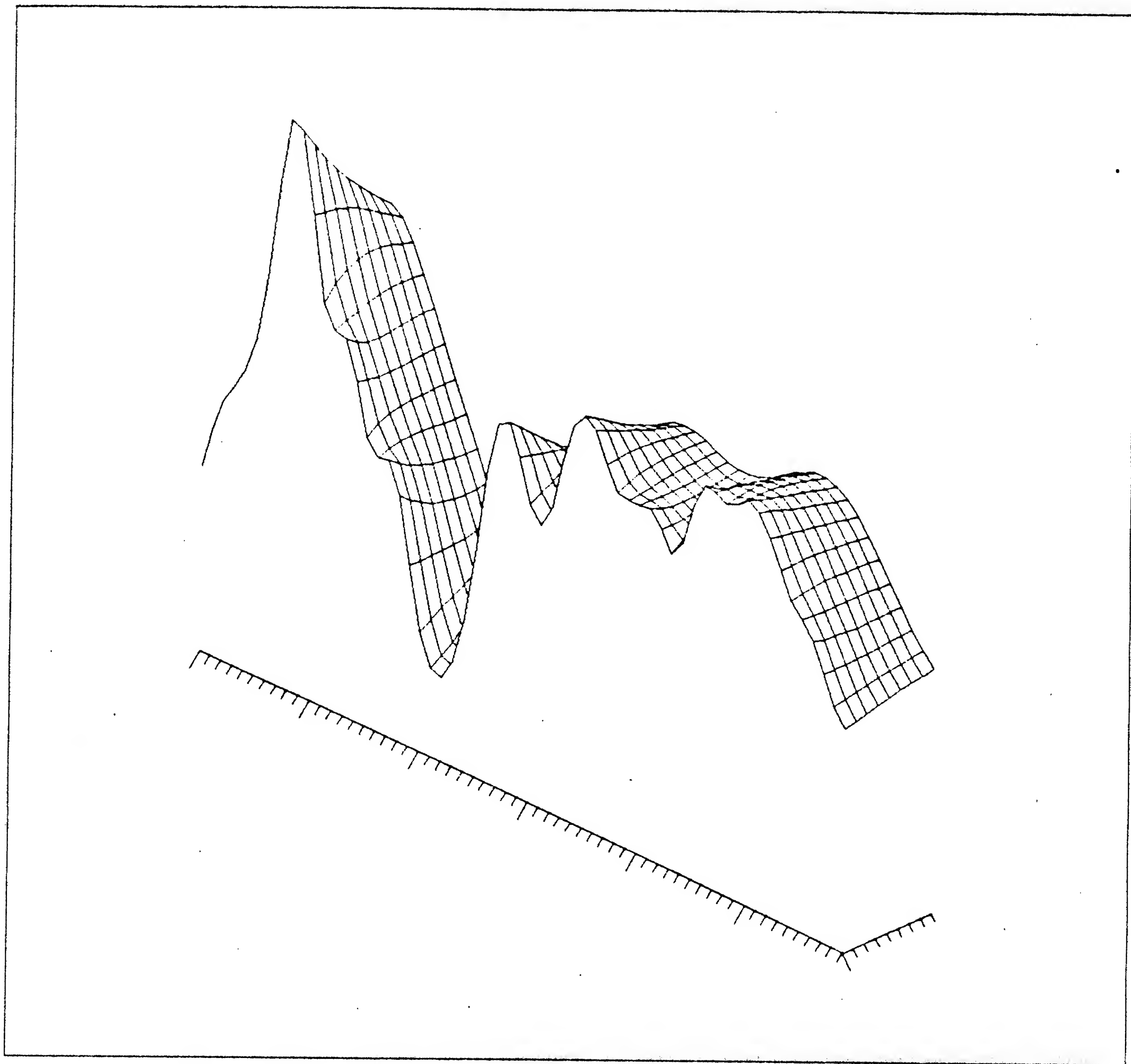
This gaussian filter is a good choice of smoothing filter for graphs of the grain market for similar reasons. Assuming that the features of the graphs are caused by events that are localized in time, the conditions fit well. The choice of degree of smoothing for a set of filters still presents difficulties. I will refer to a filter that smooths a lot as having a high sigma. Sigma is a parameter of the gaussian function. At any particular sigma there are likely to be features that have not quite smoothed away. If we increase sigma a little to erase them it usually happens that other features reach the vestigial stage. No set of sigmas returns a set of smoothed graphs each of which shows only complete features of a single scale. The point at which a feature fades out depends on the particular feature.

To attack this problem I needed to see more clearly how features gradually faded away with increased smoothing. Figure 11 shows this process for the graph of barge-shipments. A set of smoothed graphs with gradually increasing sigma are arranged side by side so that the disappearance of features is clear. In figure 11, the vertical axis represents the smoothed barge shipments, the long horizontal axis represents time, and the short horizontal axis represents the degree of smoothing, sigma.

One way of segmenting each graph in figure 11 into features is to take segments where the direction of curvature is constant, either concave or convex. The dividing lines between features then become points where the second derivative crosses zero, just as with the edges in image-processing. Figure 12 shows a similar surface for this second derivative. It is easier to look at the contour diagram of figure 13 which shows only the zero-points. Each contour is shaped like an inverted "U". The area enclosed by it represents a convex or concave segment of the surface which varies in width until, at some sigma, it is finally smoothed away. Some areas enclose others. This provides a way of setting up the hierarchy of descriptions that was desired. If the shape of the "U"s are described, perhaps using their width and depth, and the enclosure relationships between them are represented, this would provide a description of the some features of the graph that is independant of any particular choice of smoothing level.

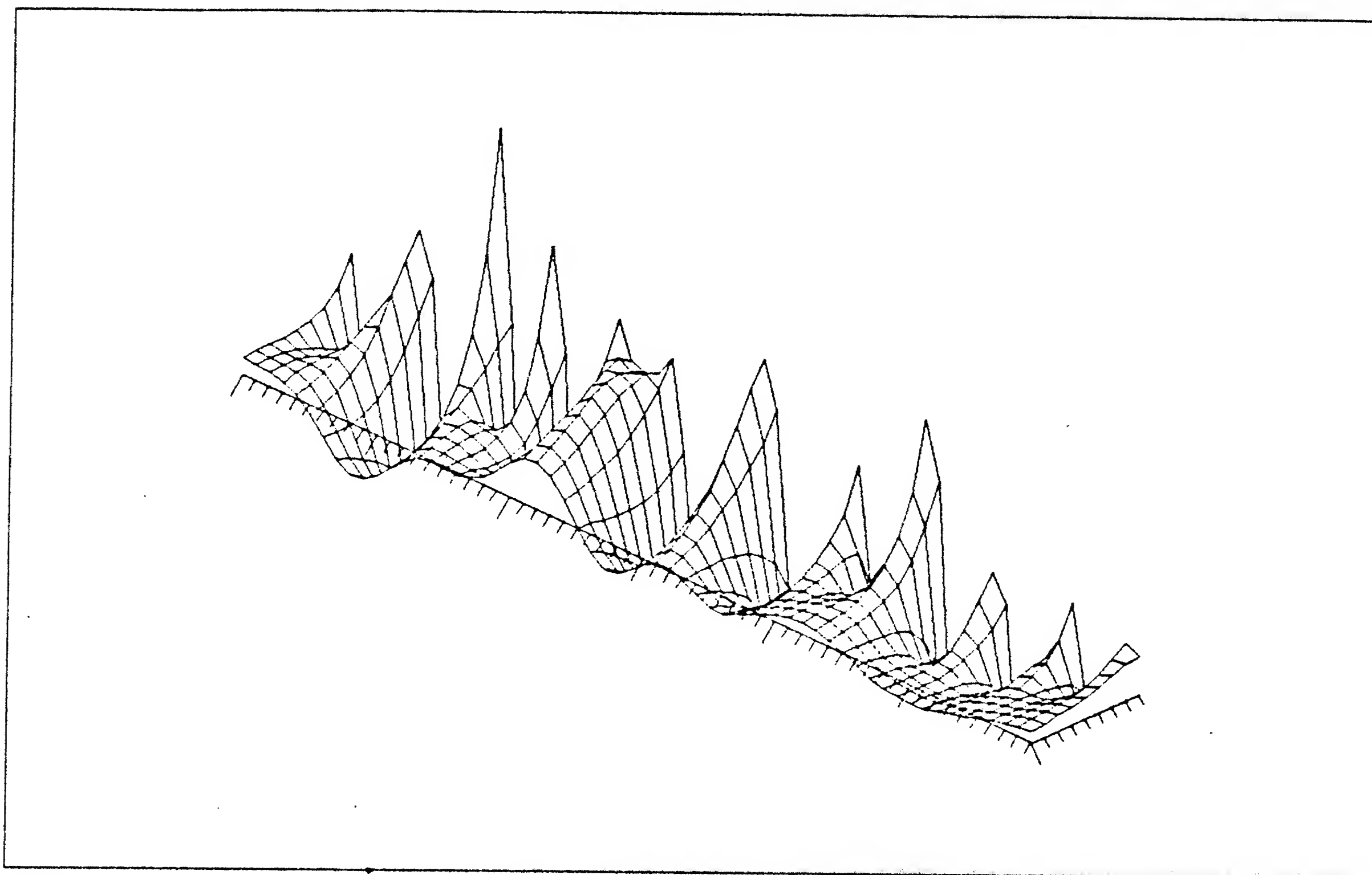
I put considerable effort into this attempt at describing features and although some progress was made I became increasingly aware that I needed to know which features an expert would find relevant. Since I had no commodity expert to consult, this was not possible. My own efforts to

Fig. 11. Gradually smoothing a graph.



construct realistic arguments about commodity graphs had brought limited success, and therefore shed no light on the features that should be extracted. In order even to experiment with different kinds of segmentation methods, I needed programs for manipulating graphs and their various functions, for displaying them, and for managing a data base of related graphs. Writing these support programs was time-consuming in itself. Because of problems like these, work on the graph description phase was eventually abandoned.

Fig. 12. The second derivative at various levels of smoothing.



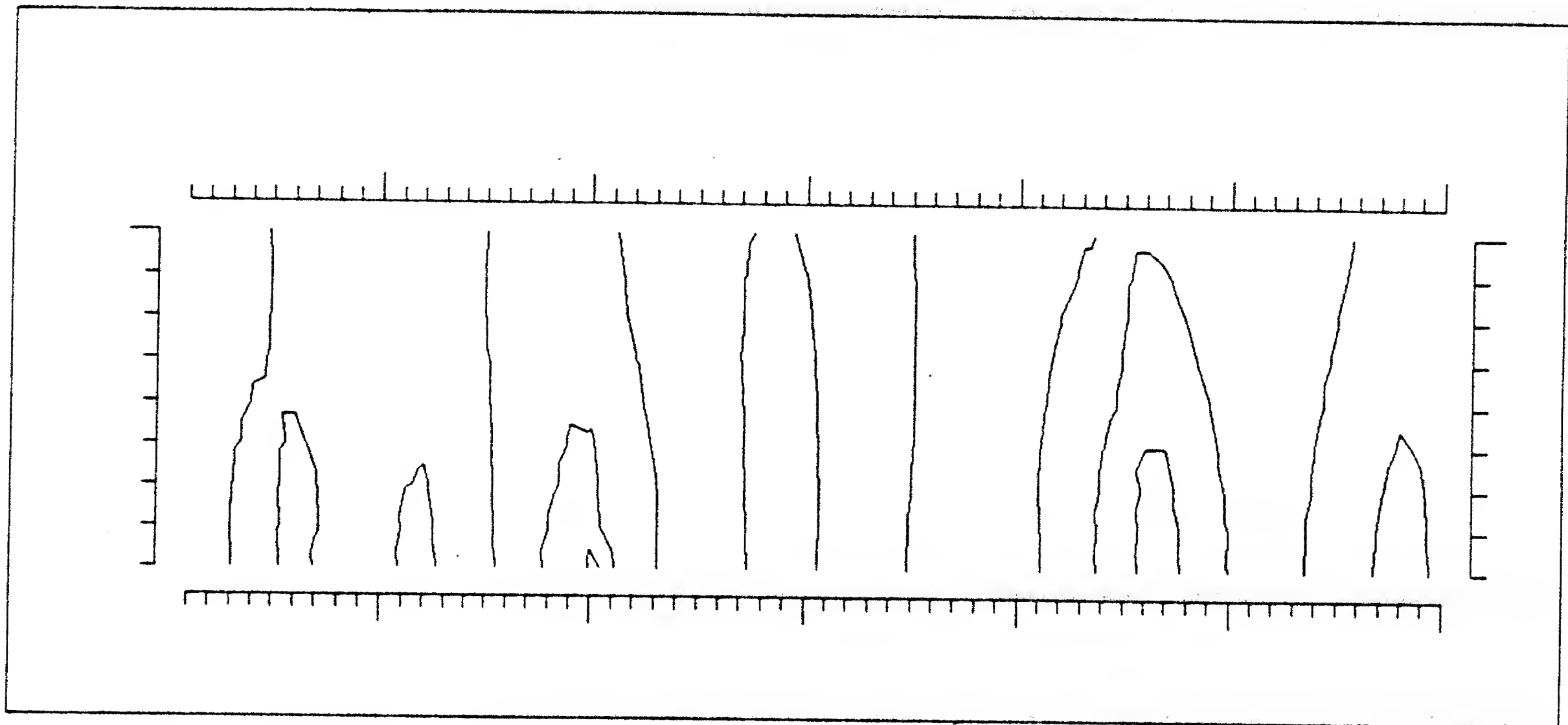
---

### Conclusion.

I have discussed three approaches taken in my work on a prototype commodity assistant, and described some of the problems with each. On looking back, it is clear to me that the project was too ambitious in its original form, but the difficulties were not so evident in the early stages. The complexity of a real-world domain can be very deceptive. Another problem was continual uncertainty about the direction the work should be developed. I think a distinction should be made between two very different kinds of project in order to ensure that expected results are achieved. One type is intended to produce an expert system for some application domain. The other uses the domain only as a source of examples and problems for research into fundamental areas such as representation, reasoning and language. Because the two kinds of project require very different methodologies, the distinction is important. Projects of the first kind should be limited in scope, and very clearly defined. While they do not call for general purpose intelligent systems they will



Fig. 13. Contours of the zero-crossings at various smoothing levels.



benefit from artificial intelligence techniques. Whatever domain is chosen, it is important to have access to an expert in that area. The program's design will depend heavily on the tasks this expert can accomplish and the methods the expert uses. The other type of project, oriented toward more fundamental research issues, need not be so closely tied to the domain. Drastic simplifications are perfectly in order if they clarify a particular issue. I think it will be quite a long time before the results of such fundamental research develop into theories that can be applied to the goals of the first but the groundwork is essential. The major misconception in the commodity expert project was the idea that this could be done sufficiently well already.

## References

- Bobrow, D. G., & Winograd, T., KRL: A knowledge representation language, *Cognitive Science* 1 (1) (January 1977) 1-46.
- Brachman, R. J., A structural paradigm for representing knowledge, (Technical Report 3605) Bolt, Beranek and Newman Inc., Cambridge MA, 1978.
- Buchanan, B. G., Sutherland, G. L., & Feigenbaum, E. A., Toward an understanding of information processes of scientific inference in the context of organic chemistry, in Meltzer, B., & Michie, D., eds., *Machine Intelligence* 5 (Edinburgh University Press, Edinburgh), 1970.
- Cullingford, R., Script application: computer understanding of newspaper stories, (Research Report 116) Yale University Department of Computer Science, 1978.
- de Kleer, J., Qualitative and quantitative knowledge in classical mechanics, (Technical Report 352) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1975.
- de Kleer, J., Doyle, J., Steele, G. L. Jr., & Sussman, G. J., Explicit control of reasoning, (Memo 427) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1977.
- de Kleer, J., Doyle, J., Rich, C., Steele, G.L., & Sussman, G. J., AMORD: A procedural deduction system. (Memo 435) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1978.
- Forrester, J. W., Industrial Dynamics (M.I.T. Press, Cambridge MA) 1965.
- Hendrix, G. G., Encoding knowledge in partitioned networks, in N. V. Findler, ed., *Associative networks - the representation and use of knowledge by computers*, (Academic Press, New York NY) 1979.
- Hieronymus, T. A., Economics of Futures Trading, (Commodity Research Bureau, New York NY) 1971.
- Marcus, M. P., A theory of syntactic recognition for natural language, (M.I.T. Press, Cambridge MA), 1980.
- Marr, D., & Hildreth, E., The theory of edge detection, (Memo 513) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1979.
- Martin, W. A., Roles, co-descriptors, and the formal representation of quantified english expressions. (Technical Report 139) M.I.T. Laboratory for Computer Science, Cambridge MA, 1979.

McDermott, D., & Sussman, G. J., The Conniver reference manual, (Memo 259a) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1974.

Minsky, M., A framework for representing knowledge, in P.H. Winston, ed., *The Psychology of Computer Vision*, (Mcgraw Hill, New York NY) 1975.

Minsky, M., Plain talk about neurodevelopmental epistemology, *Proc. 5th International Joint Conference on Artificial Intelligence*, 1977.

Meadows, D. L., Dynamics of Commodity Production Cycles, (Wright-Allen Press, Cambridge MA) 1970.

Pugh, A. L., III, DYNAMO User's Manual, 2nd ed., (M.I.T. Press, Cambridge MA) 1963.

Raulerson, R. C., & Langham, M. R., Evaluating supply control policies for frozen concentrated orange juice with an industrial dynamics model, in Labys, ed., *Qualitative models of commodity markets*, (Ballinger, Cambridge MA) 1975.

Rieger, C., & Grinberg, H., The causal representation and simulation of physical mechanisms, (Technical Report 495) University of Maryland, 1976.

Roberts, R. B., & Goldstein, I. P., The FRL Manual, (Memo 409) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1977.

Rosenberg, S. R., Discourse structure, (Working paper 130) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1976.

Rosenberg, S. R., Understanding in incomplete worlds, (Memo 475) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1978.

Schank, R., & Abelson, R., Scripts, plans goals and understanding (Lawrence Erlbaum Associates, Inc., Hillsdale NJ) 1975.

Schubert, L. K., Extending the power of semantic nets, *Artificial Intelligence Journal* 7, (1976) 163-198.

Shortliffe, E. H., MYCIN: Computer-based consultations in medical therapeutics, (American Elsevier, New York NY), 1976.

Sidner, C. L., Towards a computational theory of definite anaphora comprehension in english discourse, (Technical Report 537) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1979.

Stansfield, J. L., COMEX: A support system for a commodities analyst, (Memo 423) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1977.

Stansfield, J. L., A test-bed for developing support systems for information analysis, *Proc. 2nd National Conference CSCSI*, Toronto, 1978.

Stockman, G., Kanal, L., & Kyle, M. C., Structural pattern recognition of carotid pulse waves using a general waveform parsing system, *Communications of the A.C.M* 19 12, 1976

Sussman, G. J., Winograd, T., & Charniak, E., Micro-Planner reference manual, (Memo 203a) M.I.T. Artificial Intelligence Laboratory, Cambridge MA, 1971.

Webber, B. L., A formal approach to discourse anaphora, (Technical Report 3761) Bolt, Beranek and Newman Inc., Cambridge MA, 1978

Woods, W. A., Kaplan, R. H., & Nash-Webber, B. L., The lunar sciences natural language information system: Final report, (Technical Report 2378) Bolt, Beranek and Newman Inc., Cambridge MA, 1972.

Woods, W. A., What's in a link: foundations for semantic networks, in D. G. Bobrow, and A. M. Collins, eds., *Representation and understanding: Studies in cognitive science*, (Academic Press, New York NY) 1975.